

Received 19 October 2023, accepted 14 November 2023, date of publication 16 November 2023,
date of current version 22 November 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3333895

RESEARCH ARTICLE

A Multistage Hybrid Deep Learning Model for Enhanced Solar Tracking

MWENGE MULENGA¹, MUSA PHIRI², LUCKSON SIMUKONDA², AND FADELE A. ALABA³

¹Business Studies Division, National Institute of Public Administration, Lusaka 31990, Zambia

²School of Engineering and Technology, Mulungushi University, Kabwe 80415, Zambia

³Department of Computer Science, Federal College of Education, Zaria, Kaduna 3045, Nigeria

Corresponding author: Mwenge Mulenga (mwenge2008@yahoo.co.uk)

ABSTRACT Solar tracking helps maximize the efficiency of solar applications, such as photovoltaic (PV) solar panels. In the recent past, machine learning (ML) techniques have been extensively used to implement automatic solar tracking. However, applying predictive models in solar trackers is a non-trivial task due to the randomness and non-linearity of meteorological data, limiting their ability to clearly represent the underlying data patterns. Most existing predictive models take a monolithic approach to addressing limitations related to meteorological data, thereby limiting their performance. Therefore, this paper proposes a deep hybrid learning (DHL) model to enhance solar tracking performance. Furthermore, the proposed model improves feature representation in the data by using combined normalization methods and conversion of numerical data to images. In a nutshell, the model integrates sine and cosine transformations (SCT) to reveal cyclical patterns in the data, sigmoid and minimum-maximum data transformations to scale the data to a Gaussian distribution, and Gramian Angular Fields (GAF) to convert tabular data into 2D image representation to take advantage of the feature extraction capability of a convolutional neural network (CNN). The model also utilizes long-term short-term memory (LSTM) and gated recurrent units (GRU) for both spatial and temporal feature extraction. The results show that the aggregation of the above-mentioned methods significantly enhances solar tracking. The proposed hybrid model outperforms existing methods on a publicly available dataset, achieving outstanding performance with MAE, MAPE, and RMSE scores of 0.0073, 1.4635, and 0.0097, respectively.

INDEX TERMS Clean energy, data normalization, feature extraction, deep learning, solar tracking.

I. INTRODUCTION

In recent years, the world's energy requirements have surged, leading to increased costs in power generation [1]. To combat the negative economic and environmental impact of conventional energy sources, the focus has shifted to renewable energy. Solar energy has gained popularity due to its abundance and environmental friendliness. It is estimated that the solar energy incident on Earth's surface annually is nearly 10,000 times the world's current energy consumption [2]. At present, PV solar panels, dating back to the mid-20th century, remain the most effective technology for harnessing the sun's energy [3]. Research has shown that the amount of sunlight captured by PV solar panels is

maximized when the sun's rays are incident to the surface of the panels [4]. One way in which this can be achieved is by using a solar tracker which is able to track the position of the sun and maintain an optimum angle between the sun's rays and the surface of the PV panels [5]. However, factors such as unpredictable weather patterns and the non-linear movement of the sun across the sky have a significant impact on the performance of a solar tracker in tracing the position of the sun [1], [6]. Therefore, based on this background, reducing solar tracking errors has become a prominent topic in renewable energy [7].

Over the years, various methods have been proposed to improve the performance of solar tracking systems. These methods include, astronomical-based models [8], [9], conventional machine learning (ML) models [10], [11], and more recently, deep learning (DL) models [12], [13].

The associate editor coordinating the review of this manuscript and approving it for publication was Roberto C. Ambrosio¹.

Astronomical-based models use mathematical equations based on astronomical data. These models offer a significant level of simplicity, reliability, and fast computation [14]. However, these models are limited in their ability to capture the underlying relationships between metrological features, especially during days of cloud cover, thereby affecting their prediction accuracy [8], [15], [16]. In contrast, conventional ML models can capture non-linear relationships, offer fast computation, and high accuracy at a low-cost [17]. However, conventional ML-based models are limited in their ability to extract deeper features from data and require extensive manual data preprocessing for optimal performance [18], [19]. Fortunately, DL models have emerged as a superior approach, overcoming the limitations of shallow learning models [20]. DL models use a distributed and hierarchical feature representation method, which allows them to automatically extract deep hidden features and relationships in the data. This excellent feature extraction capability has led to the adoption of DL models in various fields of solar energy forecasting [21], wind speed prediction [22], and electrical load prediction [23], making them a promising choice for solar tracking systems.

DL-based models such as convolutional neural networks (CNN) and recurrent neural networks (RNN) such as long short-term memory (LSTM) and gated recurrent units (GRU) have been widely used to enhance solar tracking systems [24]. However, each of these models has its own individual constraints. For instance, CNN models perform exceptionally well when learning features of image-based data. However, metrological data is often represented in one dimension (1D). This means that when leveraging the CNN's feature extraction capabilities, the kernels can only move in a single direction, this constrains the CNN's ability to learn deep spatial relationships in the data [20]. Similarly, LSTM and GRU models thrive in capturing time-bound temporal dependencies but may not be sufficient for capturing spatial features in the data [25]. Various studies have noted that the integration of the CNN, LSTM, and GRU models has the potential to improve their performance compared with the corresponding individual models [26], [27], [28]. However, most of these studies rely on 1D CNN models for spatial feature extraction of metrological data.

Moreover, other studies have also shown that the choice of data normalization methods has a significant effect on the performance of the DL models [24], [29], [30]. Data normalization mitigates the influence of dominant features on the learning algorithm and reshapes the data distribution to facilitate the extraction of underlying relationships [29], [31]. Presently, existing DL-based methods aim to enhance predictive performance by employing single method-based data normalization approaches [32], [33]. One commonly used data normalization method is minimum-maximum normalization (MMN), which scales dataset values to a range from 0 to 1 while preserving the original relationships [29]. However, MMN is sensitive to extreme values and lacks

the adaptability to varying minimum and maximum feature values over time [34], [35]. Alternatively, the sigmoid normalization (SN) method, which utilizes a sigmoid function, normalizes features independently of the data distribution, making it suitable for unknown data distributions during model training [31]. Nevertheless, recent studies have shown that the aggregate of normalization methods has a positive impact on the performance of DL models [36], [37]. However, studies combining aggregate data normalization and spatial-temporal feature extraction to enhance solar tracking are limited.

In this study, we propose a model that combines aggregate data normalization, tabular to 2D image conversion, and spatial-temporal feature extraction to enhance solar tracking. The model uses an integration of the SN and MMN methods. Furthermore, the model leverages the spatial feature extraction capabilities of CNNs by converting the tabular representation of metrological data into a two-dimensional (2D) data representation. This allows the CNN kernels to move in two different directions, providing the capability to extract fine-grained feature patterns [20]. Additionally, the LSTM and GRU models are combined to improve the extraction of long short-term temporal dependencies in the data for the final prediction of the sun's position.

To achieve this, the proposed model integrates four different parts to perform accurate solar tracking. In the initial part, the sine and cosine transformation (SCT) is used to capture the oscillatory pattern in the data. Thereafter, the combination of the SN and MMN methods is performed to adjust the underlying distribution of the data toward the normal distribution. Then, the Gramian Angular Fields (GAF) method is used to convert the output from the aggregation of normalization methods into 2D images. In the subsequent step, a CNN module learns the underlying features of the images. The extracted features are then fed into the stacked LSTM and GRU layers which capture the time-bound spatial-temporal features of the data using the gating units and memory cells. The final output provides a prediction of the sun's trajectory. The proposed model is trained using a solar positioning dataset and evaluated against other standalone base models. Furthermore, the model's performance is also evaluated on both 1D and 2D variations of the dataset. Additionally, the model is compared with other published works. The experiments reveal that a 2D conversion of metrological data along with the combination of aggregate data normalization and spatial-temporal feature extraction provides the best results. This shows that a 2D representation of the metrological data reveals more hidden patterns.

Thus, our contributions in this paper are presented as follows:

- A novel aggregate data normalization approach comprising two base normalization methods, namely, SN and MMN, in which original input data is first rescaled using the former base normalization method and its

output transformed using the later base normalization method.

- A novel model that combines the aforementioned aggregate normalization, SCT, GAF, CNN, LSTM, and GRU modules to enhance solar tracking.

The remainder of this paper is structured as follows. The second section provides a summary of pertinent literature. The proposed approach is explained in the third section. The findings from the study are covered in Section Four. Section Five offers a discussion of the study's findings. The conclusion and recommendations for future research are presented in Section Six.

II. LITERATURE REVIEW

Recent advancements in artificial intelligence have seen a rise in the application of DL methods to optimize the performance of solar tracking systems. Data normalization techniques have been extensively applied in improving the predictive performance of DL methods using renewable energy-based datasets [29]. For instance, Al-Muswe et al. [38] used the z-score (ZS) normalization method to scale their data. The authors reported a noticeable improvement in their model's predictive performance. However, while ZS normalization can adequately handle outliers, its inability to transform feature values into a common numerical range may limit the performance of a predictive model. Similarly, Oviedo et al. [39] used the MMN method to transform their data to a range between 0 and 1 before feeding it into a DL model. The DL model was able to accurately learn the underlying patterns of the data partly due to the effect of MMN. However, the normalization method sometimes fails to efficiently handle outliers. Additionally, Tasdemir et al [40] used the SN method which is suitable for handling the effects of outliers. Nevertheless, the SN method's approach to handling outliers may lead to information loss and poor performance of the learning algorithm. Furthermore, literature has shown that no single normalization technique is superior to the others across all datasets as datasets have distinct underlying characteristics, which restricts the generalizability of normalization methods [36].

Given the time-series-based nature of solar tracking and its tabular data representation, most researchers use RNN and LSTM models to track the sun's position as they are well suited for sequential data. For example, Haris et al. [41] used tabular data to train an RNN model for optimized solar tracking. The study's results showed that the RNN model could accurately predict the sun's position. However, while the model was able to capture the temporal dependencies in the data, the authors did not account for the cyclical behavior in the data, which may have limited the model's performance. Likewise, Kaul et al. [42] used an LSTM model to develop an adaptive solar tracking system. Although the LSTM was able to improve the performance of the model due to its ability to capture long-term dependencies in the data, the authors did not address the issue of spatial features. In contrast, other researchers have sought to leverage the power of CNNs

using image-based datasets. For example, Pierce et al. [43] proposed a multi-input CNN that uses sky image data to predict the trajectory of the sun. Though the model was able to effectively track the sun, the authors did not take into consideration the effect of seasonal patterns on the predictive performance of the model.

Due to the drawbacks of single-based DL models, there has been a growing interest in deep hybrid learning (DHL) models among researchers. DHL models can automatically identify and combine various latent features to enhance predictive performance [44]. For instance, Syahram & Effendy [45] used an RNN-LSTM hybrid model which accounted for both the short-term and long-term dependencies in the data to forecast the sun's trajectory. However, the authors did not address the impact of spatial features which may have limited the performance of their model. In another study, Frizzo Stefenon et al. [46] integrated the wavelet energy coefficient (WEC) and LSTM. The WEC was used for signal pattern extraction, while the LSTM performed time-series-based forecasting. While this approach significantly improved the predictive performance of the model, the authors did not address the importance of spatial patterns in the data.

Furthermore, Lee et al. [47] used a CNN-LSTM model to forecast 24-hour ahead solar power. The model showed remarkable performance as the CNN was able to extract short-term local features while the LSTM captured long-term features. However, unlike image data, the tabular data used in the study may have limited the CNN module's ability to extract deeper features. Likewise, J. Wang et al. [17] proposed a DHL model for thermal power forecasting. In their study, the CNN module was used to extract spatial-temporal features, the LSTM was used to learn the temporal dependencies, and the multi-layer perceptron (MLP) for forecasting. While the model was able to accurately predict thermal power, the authors did not apply any data normalization which may have negatively affected the performance of the DHL model.

Therefore, considering the gaps identified in the reviewed works, we implement an algorithm that integrates four techniques, namely, two aggregated data normalization methods, cyclic transformations, image conversion, and feature extraction (short-term and long-term feature extraction as well as spatial feature extraction) to enhance the predictive performance of the DL model. This study employed data normalization techniques that scaled the original features using the SN and MMN methods. Combining the individual strengths of the SN and MMN methods allows for an enhanced form of shifting the data's underlying distribution toward the Gaussian distribution [36]. Furthermore, the cyclic transformations used are based on the SCT, which helps in revealing the oscillatory patterns in the data. Additionally, the spatial feature extraction used is based on a CNN, which is commonly employed in image processing tasks. Finally, the LSTM and GRU are used to extract long-term and short-term dependencies, respectively.

III. METHOD

This section provides a description of how the aggregate data normalization and SCT-GAF-CNN-LSTM-GRU modules are combined to formulate an enhanced DL-based solar tracking model. The section begins by detailing the dataset used in this study, followed by an outline of the underpinning concepts of the SCT, SN, MMN, GAF, CNN, LSTM, and GRU modules. Additionally, all the experiments were carried out using the Python 3.9-based Keras framework with Tensorflow [48]. The program was implemented using a Graphical Processing Unit (GPU) and 12 GB of RAM on Google Collaboratory [49]. Furthermore, we also leveraged the Nvidia CUDA Deep Neural Network (cuDNN) toolkit, a powerful library that provides GPU functionality for DL models to optimize model performance [50].

A. DATASET DESCRIPTION

This study used a real-world-based dataset from the Girsol sky imaging and global solar irradiance repository [51], [52]. The features included in the experiment were UNIX time, temperature ($^{\circ}\text{C}$), atmospheric pressure (mmHg), relative humidity (%), solar radiation (W/m^2), elevation angle, dew point ($^{\circ}\text{C}$), wind direction (radians), and wind velocity (mile/s),). These features were selected based on seasonal, weather, and environmental conditions. The evaluation angle of the sun was considered as the output variable of the model. Furthermore, the features contained in the dataset were collected over 272 days of the solar cycle from 2017 to 2019. Observations of the sun's position were collected four to six times per second, whereas the metrological data was recorded at 10-minute intervals. However, to ensure that all the features were observed at the same interval, the metrological data was interpolated to match the time resolution of the sun position data. This was effectively done using the averaging method which readjusted the data resolution to one-minute intervals.

B. SINE AND COSINE TRANSFORMATIONS

Renewable energy-based datasets, particularly those based on solar and wind energy, often have non-linear fluctuations due to weather patterns [53]. These patterns often have daily, seasonal, and yearly variations, making it difficult to predict future trends based on past patterns alone. To address this challenge, researchers in signal processing and engineering fields commonly use SCT for time series and circular analysis [54], [55]. Using SCT plays a significant role in analyzing and modeling complex renewable energy-related data. Conceptually, the SCT module is expressed by (1) and (2).

$$\sin(2\pi\phi m_i) = \sin(2\pi m_i/T) \quad (1)$$

$$\cos(2\pi\phi m_i) = \cos(2\pi m_i/T) \quad (2)$$

where, m is the outcome of interest and T represents the unit of analysis. For instance, if the unit of analysis is monthly periodicity, then $T = 12$. Likewise, if the unit of analysis is an annual cycle, then $T = 365.25$. Therefore, in this study,

the SCT module was used to extract daily, weekly, monthly, and yearly periodicity from the UNIX time (seconds) feature. Similarly, elevation angle and wind direction features were transformed into a 2-dimensional feature space by replacing each cyclic feature X with two features, cosine (X) and sine (X).

C. ADOPTED DATA NORMALIZATION METHODS

The performance of most learning algorithms is highly influenced by the choice of normalization method [56]. In this study, two normalization methods were adopted. Firstly, the SN method was adopted due to its ability to handle outliers. The SN method can compress commonly occurring values to essentially the same range without compromising its ability to rescale extreme values [57]. Secondly, MMN was used given its ability to preserve feature relationships in the data [58]. The associated formulae and descriptions of the normalization methods are detailed as follows.

1) SIGMOID NORMALIZATION

This method normalizes the data using a sigmoid function which rescales the values into a range between 0 and 1 or -1 and 1. The SN is computed by (3), where the data value f of the feature F is normalized to \hat{f} by computing:

$$\hat{n} = \frac{1}{1 + e^f} \quad (3)$$

2) MINIMUM AND MAXIMUM NORMALIZATION

MMN is one of the most common approaches to data normalization which scales the features into a range between 0 and 1 or -1 and 1 [57]. The bounded range in the MMN suppresses the effects of outliers resulting in a smaller standard deviation. The Min-Max scaler is calculated as follows.

$$f_{\text{norm}} = \frac{f - f_{\min}}{f_{\max} - f_{\min}} \quad (4)$$

where f_{\min} and f_{\max} refer to the min and max values of the features, respectively.

D. GRAMIAN ANGULAR FIELD

This study uses the GAF to encode time series into image data. The GAF represents times series in a polar coordinate system where each element is the cosine of the summation of angles [59]. Given a time series $T = \{t_1, t_2, \dots, t_k\}$ of k observations, T is normalized using (5) or (6). Thereafter, the normalized data \tilde{T} is represented in polar coordinates by using a cosine function to encode the data values and the time stamp as the radius (r) with (7), where λ represents the polar coordinate value, m_i is the time stamp and V is the constant factor used to regularize the span of the polar coordinate system.

$$\tilde{t}_{-1}^i = \frac{(t_i - \max(T)) + (t_i - \min(T))}{\max(T) - \min(T)} \quad (5)$$

$$\tilde{t}_0^i = \frac{t_i - \min(T)}{\max(T) - \min(T)} \quad (6)$$

$$\begin{cases} \lambda = \arccos(\tilde{t}_i), -1 \leq \tilde{t}_i \leq 1, \tilde{t}_i \in \tilde{T} \\ r = \frac{m_i}{V}, m_i \in \mathbb{N} \end{cases} \quad (7)$$

After encoding the time series data into the polar coordinate system, a symmetric matrix called the gramian matrix is formed. This is done by using either the gramian angular summation field (GASF) or gramian angular difference field (GADF), which take a trigonometric summation or difference of angles, respectively [59]. In this study, the data is normalized using a combination of the normalization methods described in the preceding section, unlike the standard GAF method. After that, the cosine function is applied to encode the data. We used the Gramian Angular Summation Field (GASF) to transform the encoded data into a symmetric matrix. The GASF uses the encoded data to exploit the angular perspective by considering the trigonometric summation between each inverse cosine value to identify the temporal correlation within the varying time intervals [60]. The GASF is defined by (8).

$$GASF = \begin{pmatrix} \cos(\lambda_1 + \lambda_1) & \cdots & \cos(\lambda_1 + \lambda_i) \\ \vdots & \ddots & \vdots \\ \cos(\lambda_i + \lambda_1) & \cdots & \cos(\lambda_i + \lambda_i) \end{pmatrix} \quad (8)$$

E. CONVOLUTIONAL NEURAL NETWORK

A CNN is an artificial neural network characterized by a multi-layer feedforward architecture that has widely proven its superiority in extracting underlying spatial features [61]. CNNs have received wide coverage in many areas of research due to their ability to solve image recognition, object detection, speech recognition, and classification tasks [62]. Unlike other DL models, such as the deep belief network (DBN) [63], CNNs have the characteristics of weight sharing and sparse connectivity [64]. These two features of a CNN significantly reduce the parameters it needs to learn and extract different patterns. This study uses a CNN to extract the underlying spatial relationships in the metrological data to reduce the prediction error of the sun's position. The architecture of a CNN is depicted in Fig. 1 and comprises three parts: the convolutional layer, pooling layer, and fully connected layer. Further, the computation of the CNN is defined by (9).

$$map_{hj}^{rc} = g \left(\sum_u \sum_{m=0}^{m'-1} \sum_{n=0}^{n'-1} \omega_{hju}^{mn} \bullet map_{(h-1)u}^{(r+m)(c+n)} + b_{hj} \right) \quad (9)$$

where r and c represent the row and column indexes, respectively, m and n denote the convolution filter's row and column indexes, respectively. M' and n' represent the number of rows and columns of the convolution filter, respectively. u is the feature map's index in the $(h-1)$ th layer. b_{hj} represents the bias term of the j th feature map in the h th layer. ω_{hju}^{mn} denotes the value of the position (m, n) in the convolution filter that links the u th feature map in the $(h-1)$ th layer. The value of a position (r, c) in the j th new feature map in the h th layer is represented by map_{hj}^{rc} . Likewise, $map_{(h-1)u}^{(r+m)(c+n)}$

Depicts the value of the position $(r + m, c + n)$ in the u th feature map in the $(h-1)$ th layer. The activation function used in each layer of the CNN is represented by g .

In this paper, two convolutional layers are used as initial layers to process the time series converted image data. Each convolution layer has 32 filters with a kernel size of 3×3 and applies a rectified linear unit (ReLU) activation function. Additionally, max pooling is applied using a pool size of 2×2 to downsample the spatial dimensions of the feature maps received from the convolutional layers. Further, dropout is applied at a rate of 0.25 to regularize the network and prevent overfitting. The output from the pooling layer is then flattened and reshaped to fit the input requirements of the subsequent LSTM and GRU layers.

F. LONG SHORT-TERM MEMORY

LSTM is a specific variant of the RNN module used to mitigate the vanishing and exploding gradient challenge the RNN faces [65]. LSTM comprises three main gates: the input gate, forget gate, and the output gate [28]. The primary function of the three gates is to control the flow of information in and out of the LSTM cell. More specifically, the input gate determines how much data enters the memory cell. The forget gate handles the retention of information and regulates which information is eliminated from the cell. The output gate determines which part of the information is passed on to the subsequent unit. Furthermore, the LSTM cell mainly uses the tangent and sigmoid activation functions. The following equations outline the recursive computations of the LSTM cell.

$$f_s = \sigma \left(l_{xf}^W x_{(s)} + l_{hf}^W h_{(s-1)} + b_f \right) \quad (10)$$

$$i_s = \sigma \left(l_{xi}^W x_{(s)} + l_{hi}^W h_{(s-1)} + b_i \right) \quad (11)$$

$$o_s = \sigma \left(l_{xo}^W x_{(s)} + l_{ho}^W h_{(s-1)} + b_o \right) \quad (12)$$

$$g'_s = \tanh \left(l_{xg}^W x_{(s)} + l_{hg}^W h_{(s-1)} + b_{g'} \right) \quad (13)$$

$$z_s = f_s \otimes z_{(s-1)} + i_s \otimes g'_s \quad (14)$$

$$h_s = o_s \otimes \tanh z_s \quad (15)$$

where \tanh and σ denote the tangent and sigmoid activation functions, respectively. f , i , o , g , z , and h represent the forget gate, input gate, output gate, temporary memory, new memory cell, and memory block at time step s , respectively. \otimes Represents an element-wise multiplication between two matrices. s represents the time step, and W denotes the magnitude of the window. l denotes the layer weights representing input x , and b represents the bias term. The cell structure of the LSTM module is shown in Fig. 2. This study uses two LSTM layers with 64 units in the first layer and 32 units in the second layer. A dropout layer is applied after each layer with a rate of 0.2.

G. GATED RECURRENT UNIT

The GRU is another variant of the RNN model, and however, unlike the LSTM model, the GRU has fewer gates [66].

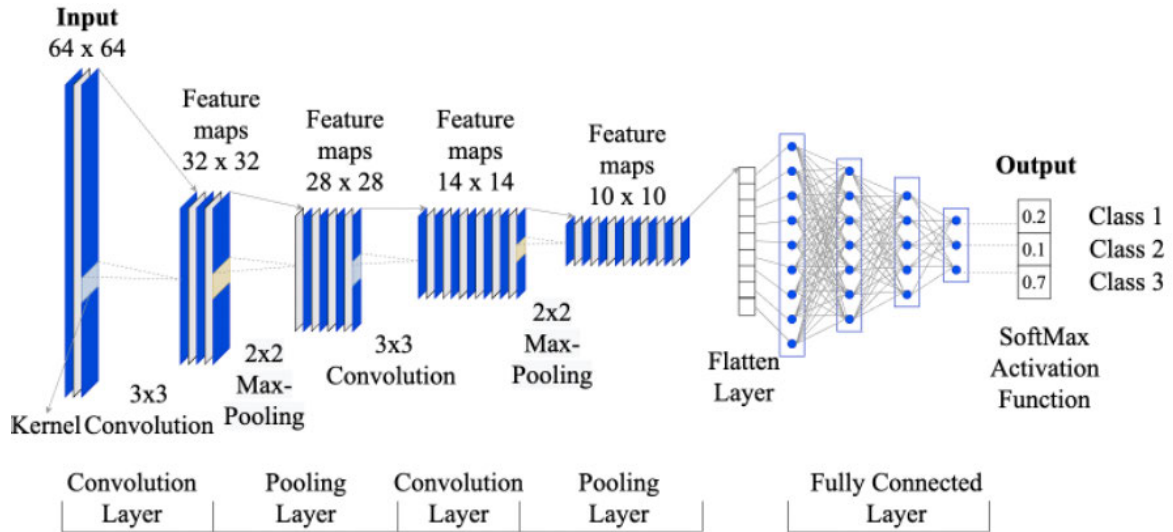


FIGURE 1. Convolutional neural network.

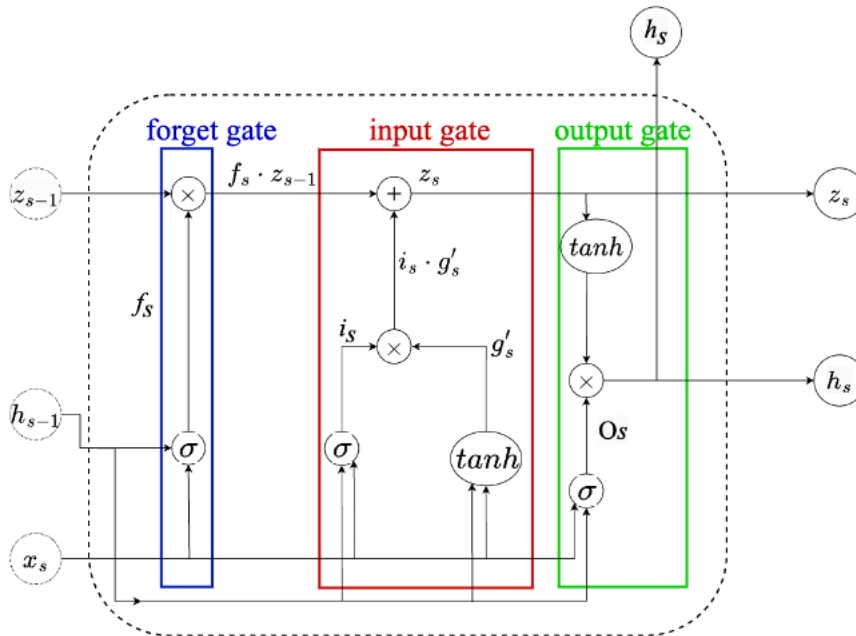


FIGURE 2. Long short-term memory.

In the GRU, the hidden state (h_t) and cell state (c_t) are controlled and merged into one [67]. Generally, the GRU model comprises two gates, the update gate (u_t) and the reset gate (r_t). The update gate (u_t) controls the extent to which the state information $h_{t-1}(c_{t-1})$ at the previous time step $t-1$ is retained in the current state t . The reset gate (r_t) determines how the information from the previous state will be integrated into the current candidate activation (\hat{c}_t). The structure of the GRU cell is shown in Fig. 3 and the update equations are defined as follows.

$$r_t = \sigma(W_{x(rg)}x_t + W_{(hs)(rg)}h_{t-1} + b_{rg}) \quad (16)$$

$$u_t = \sigma(W_{x(ug)}x_t + W_{(hs)(ug)}h_{t-1} + b_{ug}) \quad (17)$$

$$\hat{c}_t = \tanh(W_{x(cs)}x_t + W_{(hs)(cs)}(r_t \otimes h_{t-1}) + b_{cs}) \quad (18)$$

$$c_t = (1 - u_t) \otimes c_{t-1} + u_t \otimes \hat{c}_t \quad (19)$$

$$h_t = c_t \quad (20)$$

where $W_{x(rg)}$, $W_{(hs)(rg)}$, $W_{x(ug)}$, $W_{(hs)(ug)}$, $W_{x(cs)}$, and $W_{(hs)(cs)}$ are the network weight matrices. b_{rg} , b_{ug} , and b_{cs} denote the bias vectors. r_t and u_t represent the vectors of the activation values of the reset gate and update gate, respectively. This paper uses two GRU layers, with 64 units in the first layer and 32 in the second layer. Dropout at a rate of 0.2 is applied after each GRU layer.

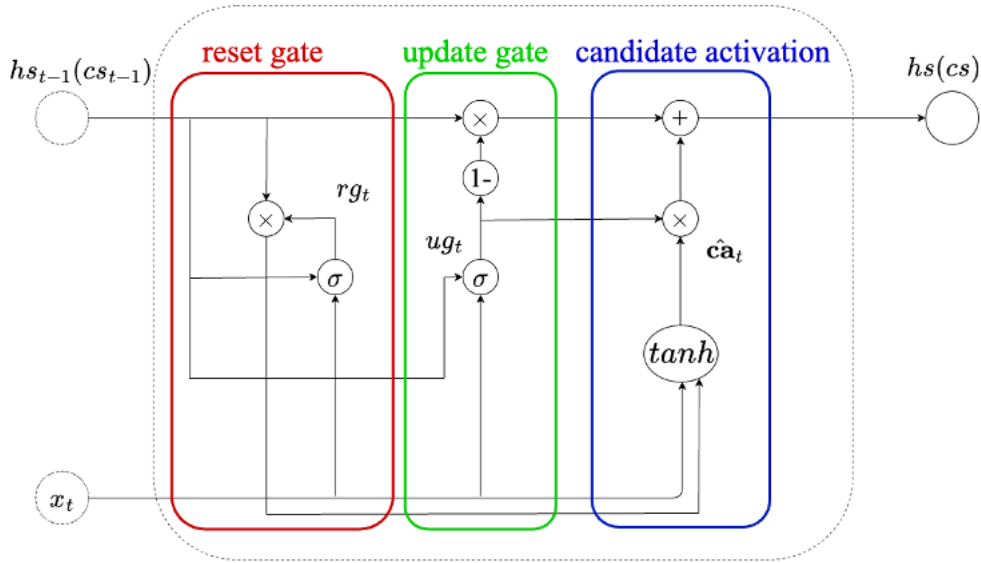


FIGURE 3. Gated recurrent unit.

H. COMBINED DATA NORMALIZATION-BASED DL MODEL

This section presents the flow of the combined data normalization-based DL model for solar tracking. The model begins by applying SCTs to the cleaned data. Thereafter, the aggregate of data normalization methods is performed. The transformed data is first normalized using the SN method. The output from the SN method is then passed to the MMN method, where the resulting output is passed to the subsequent module. The normalized data is converted to images using a GAF technique in the image conversion module. In the following module, the CNN module extracts spatial features in the image data. Thereafter, the forecasting module is used to predict the sun's position using both the LSTM and GRU models, which are ideal for identifying temporal dependencies in data sequences. The processing flow of the combined data normalization-based DL model is presented in Fig. 4.

I. PERFORMANCE METRICS

There are several performance metrics that various researchers use to evaluate the predictive ability of their models [68]. However, there is no uniform criterion for selecting the appropriate evaluation metrics [69]. Thus, in this study, we used the mean square error (MSE), mean absolute error (MAE), mean absolute percentage error (MAPE), and root mean square error (RMSE) to verify the performance of the combined data normalization-based DL model. The MSE was used as the loss function to calculate the average squared difference between the actual values k_i and the predicted values \hat{k}_i as shown in (21).

$$MSE = \frac{1}{n} \sum_{i=1}^n (k_i - \hat{k}_i)^2 \quad (21)$$

Similarly, MAE was used to calculate the average absolute variation between the actual and predicted values as defined by (22). A low MAE value is desirable for a model.

$$MAE = \frac{1}{n} \sum_{i=1}^n |k_i - \hat{k}_i| \quad (22)$$

MAPE was used to calculate the percentage difference between the actual and observed values as explained by (23). It is expected for a model to have a low MAPE value.

$$MAPE = \frac{1}{n} \sum_{i=1}^n \frac{k_i - \hat{k}_i}{m_i} \quad (23)$$

RMSE was used to calculate the square root of the mean squared difference between the actual and predicted values as computed by (24). Smaller index values of the RMSE indicate a high model performance.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (k_i - \hat{k}_i)^2} \quad (24)$$

The suitability of the selected performance metric lay in the experiment's overall purpose, which was to predict the sun's position. The MSE provides an essential model training, validation, and verification benchmark. It provides an ideal performance measure for models that predict continuous variables because of its concept of cross-entropy [70]. Likewise, MAE is the most natural measure of average error size, and, in contrast to RMSE, it provides an unambiguous measure of the average error magnitude [71]. MAPE, on the other hand, offers reliability, ease of interpretation, clarity of presentation, and utilization of all the error-related information, which make it significantly effective [72].

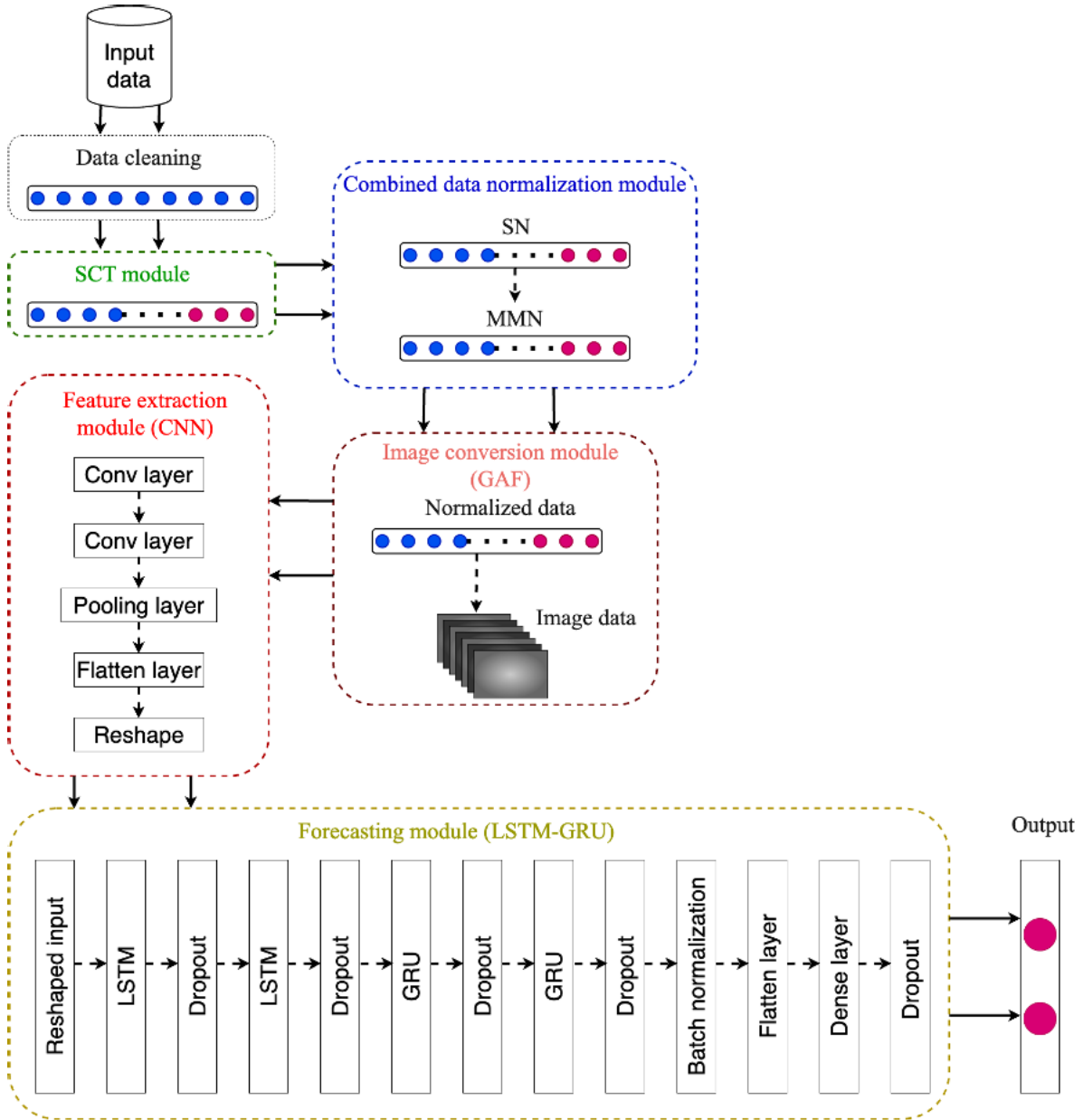


FIGURE 4. Combined data normalization-based DL model.

RMSE is a desirable error metric for several mathematical calculations because it avoids using absolute values [73]. Therefore, considering these metrics would allow for a more informed conclusion about the model’s overall performance.

IV. RESULTS AND DISCUSSION

This section analyzes the performance of the SCT-SN-MMN-CNN-LSTM-GRU model by comparing the impact of the SN-MMN and other existing data normalization methods on the model. We also demonstrate the benefit of integrating the CNN, LSTM, and GRU modules by comparing the DHL model to other DL models. Furthermore, we compare the model’s performance on both the image-transformed dataset

and the original tabular dataset. To appraise and demonstrate the merits of the SCT-SN-MMN-CNN-LSTM-GRU model, we performed 30 runs of model testing for each of the individual comparisons. Additionally, significance testing was performed to ensure that the differences observed in model performance were statistically significant.

A. COMPARISON BETWEEN DATA NORMALIZATION METHODS

To illustrate the benefit of applying the SN-MMN approach to the DHL model, we compare the model’s performance when other existing single-based data normalization methods are applied. Furthermore, we also analyzed the effect of

non-normalized (NN) data on the performance of the DHL model. This study used MMN, ZS, SN, and median scaling (MS) methods. Table 1 presents the average performance results of the model scored when each normalization method is applied.

TABLE 1. Average performance of the DHL model based on the different normalization methods.

Normalization Method	Performance Metrics		
	MAE	MAPE	RMSE
SN-MMN	0.0073	1.4635	0.0097
MMN	0.0138	2.7470	0.0193
ZS	0.0277	5.6470	0.0540
MS	0.0383	7.5295	0.0552
SN	0.0198	3.9567	0.0324
NN	0.0395	7.7230	0.0561

Based on Table 1, it is evident that the performance of the DHL model fluctuates based on the different normalization methods. Among the normalization methods, the SN-MMN method stood out with the best MAE, MAPE, and RMSE of 0.0073, 1.4635, and 0.0097, respectively, indicating exceptional predictive model performance. MMN followed with slightly worse but commendable results, recording an MAE of 0.0138, MAPE of 2.7470, and RMSE of 0.0193. SN followed, with notably higher MAE, MAPE, and RMSE scores of about 0.0198, 3.9567, and 0.0324, respectively, signaling increased prediction errors. Subsequently, ZS normalization exhibits a further deterioration in model performance, resulting in an MAE of 0.0277, MAPE of 5.6470, and RMSE of 0.0540, suggesting a relatively larger predictive error compared to the previous methods. Under the MS method, the DHL model exhibited an MAE of 0.0383, a MAPE of 7.5295, and an RMSE of 0.0552. Finally, the NN approach resulted in the worst model performance, with an MAE of 0.0395, MAPE of 7.7230, and RMSE of 0.0561, underscoring the pivotal role of normalization in enhancing predictive accuracy.

Additionally, Fig. 5 presents the scatter plot of the proposed DHL model when trained using the different normalization methods. Each normalization method is presented with two subplots, one representing the sine of the sun's elevation angle and the other representing the cosine of the angle. The vertical axes of the subplots are the predicted cosine and sine elevation angles. Similarly, the horizontal axes of the subplots are the actual cosine and sine elevation angles. Furthermore, the baselines are represented by the red diagonal line in the plots. For the SN-MMN method, the scatter plot converges well to the baseline, which demonstrates that the predicted results match the actual results. For the MMN method, there is evidence of a slight deviation between the blue points and the baseline. Also, from Fig. 5, it is indeed evident that there is a significant deviation from the baseline line for the ZS, MS, SN, and NN methods. This further justifies the phenomenal performance achieved when the SN-MMN method is used.

This may have been due to the SN-MMN method's ability to effectively scale the data to the Gaussian distribution.

B. COMPARISON BETWEEN DIFFERENT DL MODELS

In this section, we compare the performance of different DL models to the CNN-LSTM-GRU model. Specifically, we compare the proposed model to CNN, LSTM, GRU, and CNN-LSTM. This is done to analyze whether the integration of spatial feature extraction along with short-term and long-term temporal dependence extraction impacts the model's performance. Table 2 presents the average performance results of the different DL models.

TABLE 2. Average performance of the DL models.

DL Method	Performance Metrics		
	MAE	MAPE	RMSE
CNN-LSTM-GRU	0.0073	1.4635	0.0097
CNN	0.0136	2.7501	0.0194
LSTM	0.0214	4.3419	0.0252
GRU	0.0133	2.6137	0.0158
CNN-LSTM	0.0146	2.9230	0.0190

According to Table 2, the CNN-LSTM-GRU model achieved the best results with an MAE of 0.0073, MAPE of 1.4635, and RMSE of 0.0097. Following closely, the CNN model also delivered a commendable performance, scoring an MAE of 0.0136, MAPE of 2.7501, and RMSE of 0.0194. The GRU model followed with an MAE of 0.0133, a MAPE of 2.6137, and an RMSE of 0.0158, demonstrating competitive results. Subsequently, the CNN-LSTM model also showed competitive performance with an MAE of 0.0146, a MAPE of 2.9230, and an RMSE of 0.0190. Lastly, the LSTM model exhibited the lowest performance, with an MAE of 0.0214, a MAPE of 4.3419, and an RMSE of 0.0252. Furthermore, Figs. 6 and 7 present the residual plot and histogram plots of the DL models. The analysis of these residual plots and histograms is crucial for further assessing the performance and reliability of the DL models.

In Fig. 6, the subgraphs of residuals are labeled with index values on the x-axis and residuals on the y-axis. These graphs illustrate the variation between the true target values and the predicted values of the sine and cosine of the sun's elevation angles. As expected, the CNN-LSTM-GRU model displays a consistent spread of residuals across the entire range of predicted values. This indicates that the model maintains dependable performance regardless of whether predictions are high or low. Moreover, the absence of discernible patterns or trends in the differences between the model's predictions and actual values suggests that it avoids consistently overestimating or underestimating in specific situations. Similarly, the CNN-LSTM model also exhibits a relatively consistent spread of residuals and maintains a fairly random pattern. However, the absence of the GRU model may have limited its performance. Conversely, it is evident that the CNN, LSTM, and GRU models do not display random patterns, as most of the points are consistently above and below the baseline.

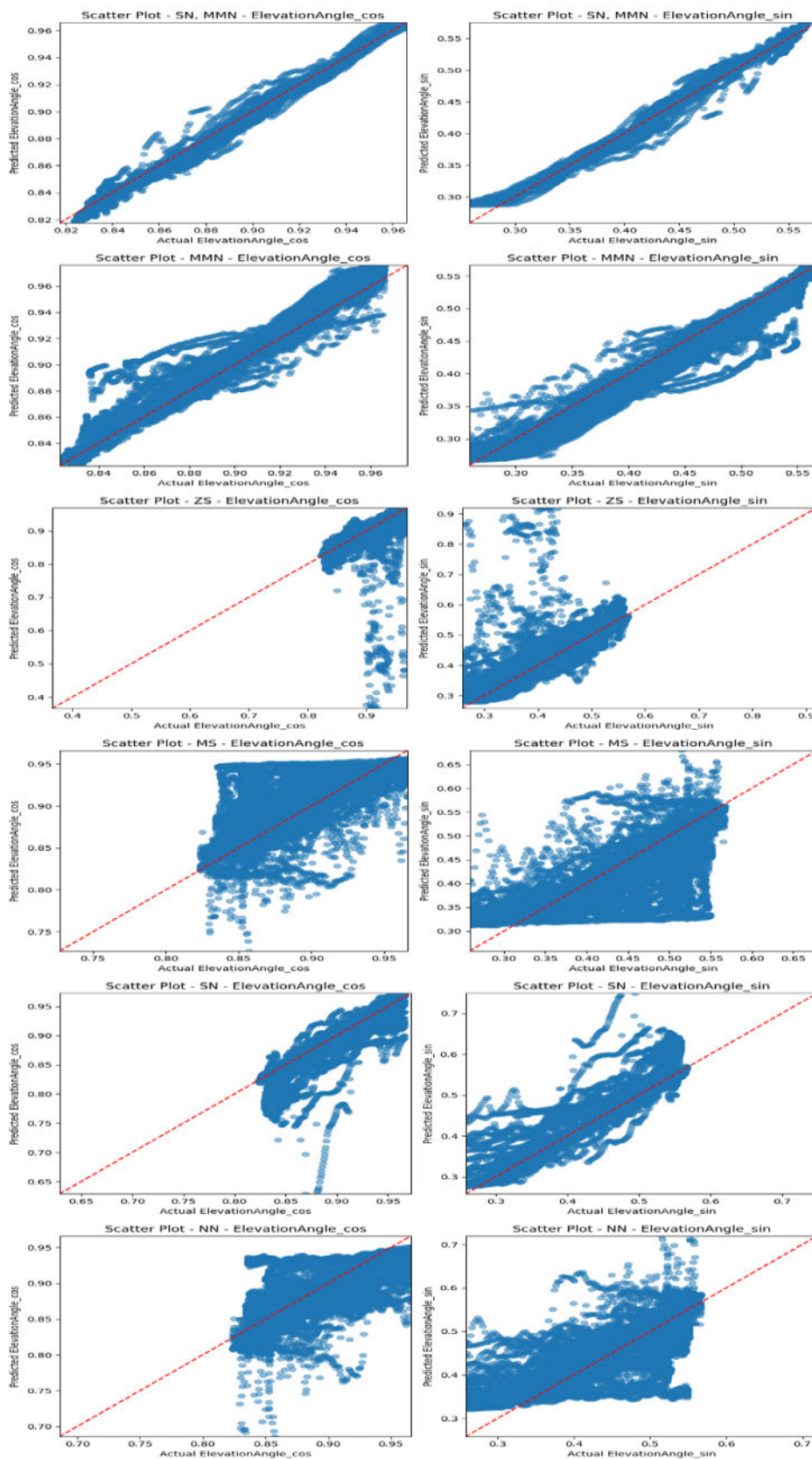


FIGURE 5. Scatter plot of predicted results.

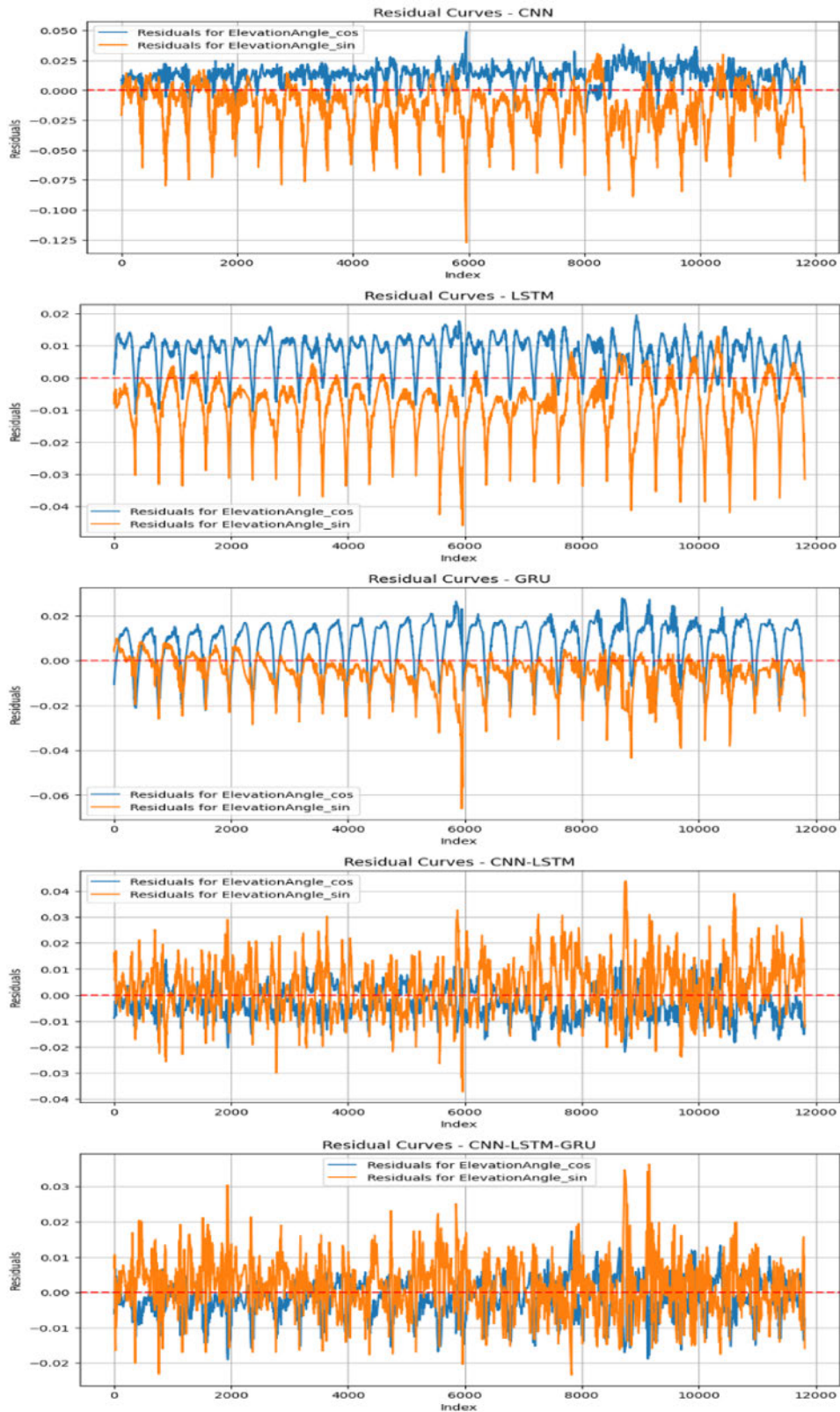


FIGURE 6. Residual graph of predicted results from DL models.

This suggests that the majority of the sun's elevation angle cosine values are systematically overpredicted by these three models. Similarly, the indication also suggests that the sine values are consistently underpredicted by the models, with this pattern being significantly evident in the case of the GRU and LSTM models.

In Fig. 7, the x-axis represents the range of residual values, and the y-axis represents the frequency of residuals within each range. It is evident that the residuals from the CNN-LSTM-GRU model form a symmetric bell-shaped curve centered near zero, implying that these residuals follow a Gaussian distribution. This observation suggests that the CNN-LSTM-GRU model effectively captures the hidden relationships within the data, demonstrating its ability to learn from the data effectively. In contrast, the other models exhibit more skewed distributions, indicating that they may struggle to capture the underlying patterns in the data accurately. Furthermore, the scarcity of unusual or extreme values in the residuals of the CNN-LSTM-GRU model indicates its capability to make reliable predictions across a diverse range of naturally occurring situations. Indeed, both the residual plot and histogram collectively highlight the superiority of the CNN-LSTM-GRU model, which is consistent with the remarkable performance observed in the MAE, MAPE, and RMSE results. Certainly, the integration of the CNN's spatial feature extraction capabilities, along with the LSTM-GRU's short and long-term temporal dependence extraction, significantly contributes to the model's outstanding performance.

C. COMPARISON OF MODEL PERFORMANCE BASED ON IMAGE AND TABULAR DATA

In this section, we compare the performance of the model on image data and tabular data. This is done to illustrate the significance of the image conversion module that is incorporated into our model. Table 3 shows the average performance results of the DHL model on image and tabular data.

TABLE 3. Average performance of DHL model using different data types.

Data Type	Performance Metrics		
	MAE	MAPE	RMSE
Image	0.0073	1.4635	0.0097
Tabular	0.0380	7.4467	0.0430

Based on Table 3, the DHL model demonstrates remarkable accuracy when trained using image data. The model achieved outstanding MAE, MAPE, and RMSE results of 0.0073, 1.4635, and 0.009, respectively. In contrast, the DHL model exhibits significantly poorer performance when trained on tabular data. The model obtained MAE, MAPE, and RMSE scores of 0.0380, 7.4467, and 0.0430, respectively. These results indicate that the image conversion module significantly enhances the model's effectiveness by revealing fine-grained relationships in the data through its conversion of tabular data into images. To further illustrate

the remarkable performance of the DHL model's image transformation module, Fig. 8 presents an analysis of the DHL model's convergence when trained with tabular data and image data.

According to Fig. 8, the loss curve of the DHL model, when trained with image data, exhibits a more rapid initial drop compared to when it is trained with tabular data. Furthermore, the DHL model trained with image data maintains a consistently lower level of loss and remains stable until the end, in contrast to the tabular data-trained DHL model. This demonstrates that the utilization of image data significantly influences the model's training, enabling it to maintain a low training loss and achieve better convergence performance. This remarkable achievement can be attributed to the DHL model's use of the 2D-CNN module, which allows it to learn deeper spatial relationships. In contrast, when the model is trained on tabular data, it is limited to using the 1D-CNN module, which restricts the number of features it can learn. Additionally, unlike the conventional GAF image transformation approach, which employs a single data normalization operation, the DHL model incorporates SN and MMN techniques. Moreover, it applies an initial SCT before image transformation and employs a cosine transformation during image transformation. These combined approaches likely contribute to the DHL model's high accuracy by leveraging fine-grained image-transformed data.

D. SIGNIFICANCE TESTING

Statistical significance testing was employed to verify that the performance of the DHL model across the different models, normalization methods, and data types was statistically significant. The model was evaluated for significance based on their MAE, MAPE, and RMSE scores. To start with, we assessed the performance results of the DHL model across the different models, normalization methods, and data types. This included determining whether the MAE, MAPE, and RMSE results followed a normal distribution. Given our small sample size, which was derived from 30 model testing runs across the different comparisons, we employed the Shapiro-Wilk test to assess the normality [74]. Using a significance level of 0.05, if the p-value exceeds 0.05, the data distribution is not significantly different from a normal distribution. Conversely, if the p-value is less than 0.05, it indicates a significant departure from normal distribution. Table 4 presents the Shapiro-Wilk normality test outcomes for the comparisons across the different models, normalization methods, and data types based on their performance results.

Based on Table 4, the Shapiro-Wilk test showed that although all the samples contained 30 elements, some of the method's performance results departed significantly from normality with p-values < 0.05 and very high statistic (W) values. Thus, due to the sample deviation from the normal distribution, the Kruskal-Wallis (H) non-parametric test was used on the samples to verify that the results from the different methods differ significantly [75]. The null hypothesis of the

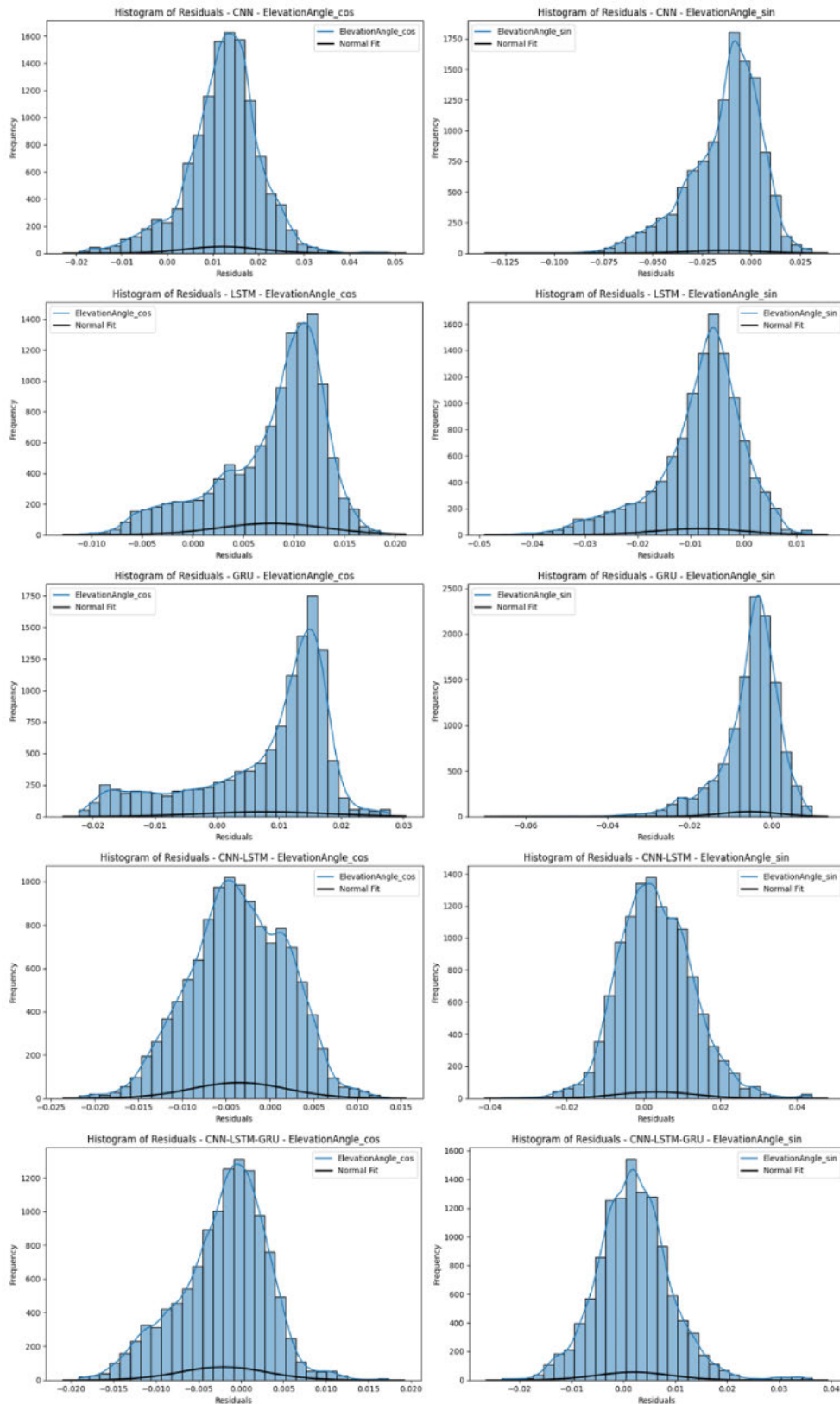


FIGURE 7. Residual histogram of predicted results from DL models.

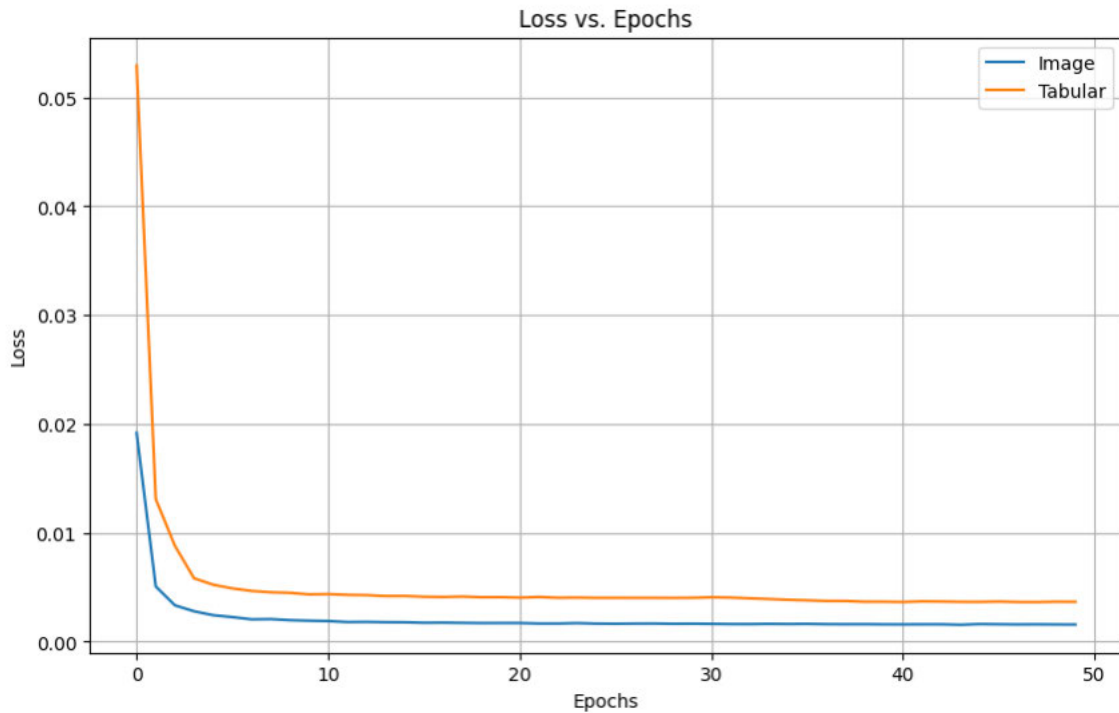


FIGURE 8. Loss curve of the DHL model with image and tabular data.

TABLE 4. Shapiro-wilk normality test results for different comparison types.

Comparison types	Methods	MAE		MAPE		RMSE	
		W	p-value	W	p-value	W	p-value
Normalization methods	SN-MMN	0.86172	0.00075	0.80845	0.00006	0.85698	0.00059
	MMN	0.81761	0.00009	0.81560	0.00008	0.95148	0.15876
	ZS	0.97537	0.65823	0.96846	0.45805	0.97547	0.66134
	MS	0.98326	0.88685	0.96221	0.31542	0.97734	0.71930
	SN	0.92985	0.03879	0.93046	0.04034	0.95577	0.20967
	NN	0.80866	0.00006	0.92054	0.02150	0.85773	0.00061
DL models	CNN-LSTM-GRU	0.81877	0.00015	0.75832	0.00001	0.81031	0.00010
	CNN	0.94108	0.09727	0.93315	0.05958	0.97936	0.80839
	LSTM	0.88393	0.00347	0.89253	0.00554	0.89905	0.00796
	GRU	0.89588	0.00667	0.89612	0.00676	0.90702	0.01253
	CNN-LSTM	0.77278	0.00002	0.74287	0.00001	0.80404	0.00008
Data types	Image	0.86172	0.00075	0.80845	0.00006	0.85698	0.00059
	Tabular	0.87701	0.00169	0.87844	0.00183	0.88114	0.00212

Kruskal-Wallis (H) test is that the data samples do not differ. If the p-value falls below the specified significance level ($\alpha = 0.05$), we reject this null hypothesis and conclude the existence of a notable distinction in medians among the groups. Therefore, the Kruskal-Wallis results for comparing the different method performance result samples are shown in Table 5.

Based on Table 5, all the p-values recorded are considerably less than 0.05, indicating strong evidence against the null hypothesis of equal medians in performance scores for the comparisons across the different models, normalization methods, and data types. Therefore, we reject the null hypothesis, suggesting that at least one of the individual

methods has a significantly different median performance score compared to the others. To further investigate the pairwise differences between the comparison types, post-hoc testing was done using Bonferroni-Dunn’s test. A two-tailed null hypothesis at the 0.01 and 0.05 level of significance was employed. Table 6 presents the results of the significance testing for the DHL model with the SN-MMN method against the other single-based methods.

As seen in Table 6, the group-wise p-value comparison between the SN-MMN method and the other normalization methods, except MMN, is statistically significant at the 0.001 level. Nevertheless, the difference observed between the SN-MMN and other methods is statistically significant

TABLE 5. Kruskal-wallis test results based on comparison types.

Comparison types	MAE		MAPE		RMSE	
	H	p-value	H	p-value	H	p-value
Normalization methods	1.79E+02	1.10E-36	1.79E+02	1.06E-36	1.64E+02	1.26E-33
DL models	6.04E+01	2.39E-12	6.10E+01	1.78E-12	6.64E+01	1.33E-13
Data types	4.73E+01	6.21E-12	4.73E+01	6.21E-12	4.73E+01	6.21E-12

TABLE 6. Post-Hoc tests results of the SN-MMN and other normalization methods.

Performance Metrics	Comparison	p-value (two-tailed)	
		($\alpha = 0.001$)	($\alpha = 0.05$)
MAE	SN-MMN vs. MMN	> 0.001	< 0.05
	SN-MMN vs. MS	< 0.001	< 0.05
	SN-MMN vs. SN	< 0.001	< 0.05
	SN-MMN vs. ZS	< 0.001	< 0.05
	SN-MMN vs. NN	< 0.001	< 0.05
MAPE	SN-MMN vs. MMN	> 0.001	< 0.05
	SN-MMN vs. MS	< 0.001	< 0.05
	SN-MMN vs. SN	< 0.001	< 0.05
	SN-MMN vs. ZS	< 0.001	< 0.05
	SN-MMN vs. NN	< 0.001	< 0.05
RMSE	SN-MMN vs. MMN	> 0.001	< 0.05
	SN-MMN vs. MS	< 0.001	< 0.05
	SN-MMN vs. SN	< 0.001	< 0.05
	SN-MMN vs. ZS	< 0.001	< 0.05
	SN-MMN vs. NN	< 0.001	< 0.05

at the 0.05 significance level. Therefore, it can be concluded that the outcomes produced by the methods are significantly different. Table 7 displays the considerable testing results for the DHL model compared to the other DL models.

TABLE 7. Post-Hoc tests results of the DHL model and other DL models.

Performance Metric	Comparison	p-value (two-tailed)	
		($\alpha=0.001$)	($\alpha=0.05$)
MAE	CNN-LSTM-GRU vs. CNN	< 0.001	< 0.05
	CNN-LSTM-GRU vs. LSTM	< 0.001	< 0.05
	CNN-LSTM-GRU vs. GRU	< 0.001	< 0.05
	CNN-LSTM-GRU vs. CNN-LSTM	< 0.001	< 0.05
MAPE	CNN-LSTM-GRU vs. CNN	< 0.001	< 0.05
	CNN-LSTM-GRU vs. LSTM	< 0.001	< 0.05
	CNN-LSTM-GRU vs. GRU	< 0.001	< 0.05
	CNN-LSTM-GRU vs. CNN-LSTM	< 0.001	< 0.05
RMSE	CNN-LSTM-GRU vs. CNN	< 0.001	< 0.05
	CNN-LSTM-GRU vs. LSTM	< 0.001	< 0.05
	CNN-LSTM-GRU vs. GRU	< 0.001	< 0.05
	CNN-LSTM-GRU vs. CNN-LSTM	< 0.001	< 0.05

As seen in Table 7, the difference between the CNN-LSTM-GRU and other DL models was statistically significant at the 0.001 significance level. Furthermore, Table 8 shows the results of the significance testing of the performance of the DHL model when trained with image data against tabular data.

TABLE 8. Post-Hoc tests results of the DHL model on image data against tabular data.

Performance Metrics	Comparison	p-value (two-tailed)	
		($\alpha = 0.001$)	($\alpha = 0.05$)
MAE	Image vs. Tabular	< 0.001	< 0.05
MAPE	Image vs. Tabular	< 0.001	< 0.05
RMSE	Image vs. Tabular	< 0.001	< 0.05

Based on Table 8, it is evident that the $p < 0.001$ for each of the comparisons between the image data and the tabular data is statistically significant. Thus, we can conclude that the performance results of the DHL model differ significantly when trained on image and tabular data.

E. COMPARISON OF THE PROPOSED MODEL WITH OTHER RECENTLY PUBLISHED MODELS

As discussed in the literature review, several recent studies have proposed various DL-based solar tracking models. These models represent some of the most recent intelligent models designed to predict the sun’s position efficiently. However, their robustness and superiority can vary based on factors like the variables used for training, preprocessing techniques, and model complexity. Therefore, this section aims to compare the most recently developed DL models for solar tracking with the DHL model proposed in this study. We consider four recent articles in this comparison to evaluate and assess the superiority and contribution of the proposed DHL model. It’s worth noting that each of the models proposed in the previous studies was trained and tested on non-publicly available datasets. To compare their performance against that of the proposed model, all the models were trained and tested on the publicly available dataset used in this study. Table 9 provides a comparison between the proposed model and the other recently published models.

TABLE 9. Comparison of the proposed model with other recently published models.

Reference	Model	MAE	MAPE	RMSE
Proposed model	CNN-LSTM-GRU	0.0073	1.4635	0.0097
Al-Muswe et al. [38]	LSTM	0.0886	0.0677	8.4934
Haris et al. [41]	RNN	0.0655	0.0474	6.3247
Kaul et al. [42]	LSTM	0.0665	0.0508	6.6629
Syahram & Effendy [45]	RNN-LSTM	0.0711	0.0538	6.9575

As depicted in Table 9, the proposed model consistently outperforms all the models in previous studies. This highlights several key factors. Firstly, it is important to note that data types may impact the performance of models. Specifically, the previous studies' models used tabular data, and similarly, this study also utilized tabular data. However, relying solely on tabular data often limits the utilization of spatial features inherent in the data. Thus, this study incorporates a tabular-to-image conversion, which enables the extraction of spatial features. Secondly, the recent studies employed both RNN and LSTM models. The use of individual RNN and LSTM models in those studies allowed them to capture short-term and long-term dependencies in the data. In contrast, this study integrates the advantages of CNN, LSTM, and GRU models, enabling the simultaneous capture of spatial and long-term relationships in the data. Finally, a noteworthy observation in the previous studies is the use of methods based on single data normalization. Conversely, this study incorporates two different data normalization methods, effectively scaling the features by leveraging the strengths of each technique. By harnessing these capabilities, the proposed model achieves superior performance compared to previous studies.

F. CONCLUSION

This study provides empirical evidence that using the SN-MMN method with cyclic transformations, image conversion, spatial feature extraction, and short-term and long-term feature extraction significantly enhances solar tracking. Our experiments demonstrate three essential aspects for achieving success. These aspects include effective feature scaling while preserving original data relationships, leveraging image representation of the dataset, and modeling spatial and temporal dynamics using CNN, LSTM, and GRU modules. Furthermore, The proposed hybrid model outperforms existing methods on a publicly available dataset, achieving outstanding performance with MAE, MAPE, and RMSE scores of 0.0073, 1.4635, and 0.0097, respectively. However, this study has limitations. Since the proposed model was only trained on data from a single geographical location, the model may require reconfiguration according to the location of its implementation. Additionally, it should be noted that the model was trained on data collected over 272 days through three years, representing approximately 25% of the period. This may limit the generalizability of the model. Thus, future work should consider training the model on a larger dataset, as well as investigating the integration of image and tabular data for improved solar tracking. It would also be interesting to combine the two data types with ensemble and other state-of-the-art methods.

REFERENCES

[1] N. Elizabeth Michael, M. Mishra, S. Hasan, and A. Al-Durra, "Short-term solar power predicting model based on multi-step CNN stacked LSTM technique," *Energies*, vol. 15, no. 6, p. 2150, Mar. 2022, doi: 10.3390/en15062150.

[2] R. A. Rajagukguk, R. A. A. Ramadhan, and H.-J. Lee, "A review on deep learning models for forecasting time series data of solar irradiance and photovoltaic power," *Energies*, vol. 13, no. 24, p. 6623, Dec. 2020, doi: 10.3390/en13246623.

[3] Y. Yao, Y. Hu, S. Gao, G. Yang, and J. Du, "A multipurpose dual-axis solar tracker with two tracking strategies," *Renew. Energy*, vol. 72, pp. 88–98, Dec. 2014, doi: 10.1016/j.renene.2014.07.002.

[4] D. Yadav, N. Singh, V. S. Bhadoria, V. Vita, G. Fotis, E. G. Tsampanis, and T. I. Maris, "Analysis of the factors influencing the performance of single- and multi-diode PV solar modules," *IEEE Access*, vol. 11, pp. 95507–95525, 2023, doi: 10.1109/access.2023.3306473.

[5] R. Ramful and N. Sowaruth, "Low-cost solar tracker to maximize the capture of solar energy in tropical countries," *Energy Rep.*, vol. 8, pp. 295–302, Nov. 2022, doi: 10.1016/j.egy.2022.10.145.

[6] M. Ali Jallal, S. Chabaa, and A. Zeroual, "A novel deep neural network based on randomly occurring distributed delayed PSO algorithm for monitoring the energy produced by four dual-axis solar trackers," *Renew. Energy*, vol. 149, pp. 1182–1196, Apr. 2020, doi: 10.1016/j.renene.2019.10.117.

[7] N. Al-Rousan, N. A. M. Isa, and M. K. M. Desa, "Efficient single and dual axis solar tracking system controllers based on adaptive neural fuzzy inference system," *J. King Saud Univ. Eng. Sci.*, vol. 32, no. 7, pp. 459–469, Nov. 2020, doi: 10.1016/j.jksues.2020.04.004.

[8] J. Fonseca-Campos, L. Fonseca-Ruiz, and P. N. Cortez-Herrera, "Portable system for the calculation of the sun position based on a laptop, a GPS and Python," in *Proc. IEEE Int. Autumn Meeting Power, Electron. Comput. (ROPEC)*. Ixtapa: IEEE, Nov. 2016, pp. 1–5, doi: 10.1109/ROPEC.2016.7830580.

[9] M. H. M. Sidek, W. Z. W. Hasan, M. Z. A. Ab. Kadir, S. Shafie, M. A. M. Radzi, S. A. Ahmad, and M. H. Marhaban, "GPS based portable dual-axis solar tracking system using astronomical equation," in *Proc. IEEE Int. Conf. Power Energy (PECon)*. Kuching, Sarawak, Malaysia: IEEE, Dec. 2014, pp. 245–249, doi: 10.1109/PECON.2014.7062450.

[10] G. Y. Kim, D. S. Han, and Z. Lee, "Solar panel tilt angle optimization using machine learning model: A case study of Daegu city, South Korea," *Energies*, vol. 13, no. 3, p. 529, Jan. 2020, doi: 10.3390/en13030529.

[11] J. Mondal and N. Mondol, "Horizontal axis solar tracking system to enhance efficiency with accuracy test by machine learning," in *Proc. Int. Conf. Mech., Manuf. Process Eng. (ICMMPE)*, 2022, pp. 1–6.

[12] N. Al-Rousan, N. A. M. Isa, and M. K. M. Desa, "Correlation analysis and MLP/CMLP for optimum variables to predict orientation and tilt angles in intelligent solar tracking systems," *Int. J. Energy Res.*, vol. 45, no. 1, pp. 453–477, Jan. 2021, doi: 10.1002/er.5676.

[13] N. Al-Rousan, N. A. M. Isa, M. K. M. Desa, and H. Al-Najjar, "Integration of logistic regression and multilayer perceptron for intelligent single and dual axis solar tracking systems," *Int. J. Intell. Syst.*, vol. 36, no. 10, pp. 5605–5669, Oct. 2021, doi: 10.1002/int.22525.

[14] M. E. H. Chowdhury, A. Khandakar, B. Hossain, and R. Abouhasera, "A low-cost closed-loop solar tracking system based on the sun position algorithm," *J. Sensors*, vol. 2019, pp. 1–11, Feb. 2019, doi: 10.1155/2019/3681031.

[15] A. R. Amelia, Y. M. Irwan, I. Safwati, W. Z. Leow, M. H. Mat, and M. S. A. Rahim, "Technologies of solar tracking systems: A review," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 767, no. 1, Feb. 2020, Art. no. 012052, doi: 10.1088/1757-899X/767/1/012052.

[16] A. Musa, E. Alozie, S. A. Suleiman, J. A. Ojo, and A. L. Imoize, "A review of time-based solar photovoltaic tracking systems," *Information*, vol. 14, no. 4, p. 211, Mar. 2023, doi: 10.3390/info14040211.

[17] J. Wang, L. Guo, C. Zhang, L. Song, J. Duan, and L. Duan, "Thermal power forecasting of solar power tower system by combining mechanism modeling and deep learning method," *Energy*, vol. 208, Oct. 2020, Art. no. 118403, doi: 10.1016/j.energy.2020.118403.

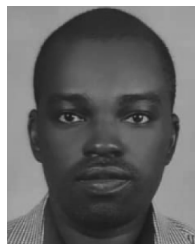
[18] N. E. Benti, M. D. Chaka, and A. G. Semie, "Forecasting renewable energy generation with machine learning and deep learning: Current advances and future prospects," *Sustainability*, vol. 15, no. 9, p. 7087, Apr. 2023, doi: 10.3390/su15097087.

[19] J.-P. Lai, Y.-M. Chang, C.-H. Chen, and P.-F. Pai, "A survey of machine learning models in renewable energy predictions," *Appl. Sci.*, vol. 10, no. 17, p. 5975, Aug. 2020, doi: 10.3390/app10175975.

[20] A. N. Sayed, Y. Himeur, and F. Bensaali, "From time-series to 2D images for building occupancy prediction using deep transfer learning," *Eng. Appl. Artif. Intell.*, vol. 119, Mar. 2023, Art. no. 105786, doi: 10.1016/j.engappai.2022.105786.

- [21] W. Khan, S. Walker, and W. Zeiler, "Improved solar photovoltaic energy generation forecast using deep learning-based ensemble stacking approach," *Energy*, vol. 240, Feb. 2022, Art. no. 122812, doi: [10.1016/j.energy.2021.122812](https://doi.org/10.1016/j.energy.2021.122812).
- [22] P. Ashok and V. Sreelekha, "Prediction of solar irradiance and wind speed using deep recurrent neural network," in *Proc. 2nd Int. Conf. Next Gener. Intell. Syst. (ICNGIS)*. Kottayam, India: IEEE, Jul. 2022, pp. 1–4, doi: [10.1109/ICNGIS54955.2022.10079733](https://doi.org/10.1109/ICNGIS54955.2022.10079733).
- [23] C. Pavlatos, E. Makris, G. Fotis, V. Vita, and V. Mladenov, "Utilization of artificial neural networks for precise electrical load prediction," *Technologies*, vol. 11, no. 3, p. 70, May 2023, doi: [10.3390/technologies11030070](https://doi.org/10.3390/technologies11030070).
- [24] M. Phiri, M. Mulenga, A. Zimba, and C. I. Eke, "Deep learning techniques for solar tracking systems: A systematic literature review, research challenges, and open research directions," *Sol. Energy*, vol. 262, Sep. 2023, Art. no. 111803, doi: [10.1016/j.solener.2023.111803](https://doi.org/10.1016/j.solener.2023.111803).
- [25] G. Tang, B. Li, H.-N. Dai, and X. Zheng, "SPRNN: A spatial-temporal recurrent neural network for crowd flow prediction," *Inf. Sci.*, vol. 614, pp. 19–34, Oct. 2022, doi: [10.1016/j.ins.2022.09.053](https://doi.org/10.1016/j.ins.2022.09.053).
- [26] H. Alharkan, S. Habib, and M. Islam, "Solar power prediction using dual stream CNN-LSTM architecture," *Sensors*, vol. 23, no. 2, p. 945, Jan. 2023, doi: [10.3390/s23020945](https://doi.org/10.3390/s23020945).
- [27] S. Ghimire, R. C. Deo, N. Raj, and J. Mi, "Deep solar radiation forecasting with convolutional neural network and long short-term memory network algorithms," *Appl. Energy*, vol. 253, Nov. 2019, Art. no. 113541, doi: [10.1016/j.apenergy.2019.113541](https://doi.org/10.1016/j.apenergy.2019.113541).
- [28] Q. Zhu, J. Chen, D. Shi, L. Zhu, X. Bai, X. Duan, and Y. Liu, "Learning temporal and spatial correlations jointly: A unified framework for wind speed prediction," *IEEE Trans. Sustain. Energy*, vol. 11, no. 1, pp. 509–523, Jan. 2020, doi: [10.1109/TSTE.2019.2897136](https://doi.org/10.1109/TSTE.2019.2897136).
- [29] G. Alkhayat and R. Mehmood, "A review and taxonomy of wind and solar energy forecasting methods based on deep learning," *Energy AI*, vol. 4, Jun. 2021, Art. no. 100060, doi: [10.1016/j.egyai.2021.100060](https://doi.org/10.1016/j.egyai.2021.100060).
- [30] P. A. C. Rocha and V. O. Santos, "Global horizontal and direct normal solar irradiance modeling by the machine learning methods XGBoost and deep neural networks with CNN-LSTM layers: A case study using the GOES-16 satellite imagery," *Int. J. Energy Environ. Eng.*, vol. 13, no. 4, pp. 1271–1286, Dec. 2022, doi: [10.1007/s40095-022-00493-6](https://doi.org/10.1007/s40095-022-00493-6).
- [31] S. C. Nayak, B. B. Misra, and H. S. Behera, "Impact of data normalization on stock index forecasting," *Int. J. Comput. Inf. Syst. Ind.*, vol. 6, pp. 257–269, Jan. 2014.
- [32] I. M. Pires, F. Hussain, N. M. M. Garcia, P. Lameski, and E. Zdravetski, "Homogeneous data normalization and deep learning: A case study in human activity classification," *Future Internet*, vol. 12, no. 11, p. 194, Nov. 2020, doi: [10.3390/fi12110194](https://doi.org/10.3390/fi12110194).
- [33] D. Singh and B. Singh, "Investigating the impact of data normalization on classification performance," *Appl. Soft Comput.*, vol. 97, Dec. 2020, Art. no. 105524, doi: [10.1016/j.asoc.2019.105524](https://doi.org/10.1016/j.asoc.2019.105524).
- [34] S. Kappal, "Data normalization using median & median absolute deviation (MMAD) based Z-score for robust predictions vs. min-max normalization," *Lond. J. Res. Sci. Nat. Form.*, vol. 19, no. 4, pp. 13110–13140, 2019, doi: [10.13140/RG.2.2.32799.82088](https://doi.org/10.13140/RG.2.2.32799.82088).
- [35] J. Pan, Y. Zhuang, and S. Fong, "The impact of data normalization on stock market prediction: Using SVM and technical indicators," in *Communications in Computer and Information Science*, vol. 652, M. W. Berry, A. H. Mohamed, and B. W. Yap, Eds. Singapore: Springer, 2016, pp. 72–88, doi: [10.1007/978-981-10-2777-2_7](https://doi.org/10.1007/978-981-10-2777-2_7).
- [36] M. Mulenga, S. A. Kareem, A. Q. M. Sabri, and M. Seera, "Stacking and chaining of normalization methods in deep learning-based classification of colorectal cancer using gut microbiome data," *IEEE Access*, vol. 9, pp. 97296–97319, 2021, doi: [10.1109/ACCESS.2021.3094529](https://doi.org/10.1109/ACCESS.2021.3094529).
- [37] M. Mulenga, S. A. Kareem, A. Q. Md Sabri, M. Seera, S. Govind, C. Samudi, and S. B. Mohamad, "Feature extension of gut microbiome data for deep neural network-based colorectal cancer classification," *IEEE Access*, vol. 9, pp. 23565–23578, 2021, doi: [10.1109/ACCESS.2021.3050838](https://doi.org/10.1109/ACCESS.2021.3050838).
- [38] A. F. A. Al-Muswe, I. K. A. Al-Amri, M. N. H. Al-Turfi, and A. Alkhayat, "Solar tracking system by utilized optimized algorithm based deep learning," in *Proc. 5th Int. Conf. Eng. Technol. Appl. (IICETA)*. Al-Najaf, Iraq: IEEE, May 2022, pp. 37–44, doi: [10.1109/IICETA54559.2022.9888545](https://doi.org/10.1109/IICETA54559.2022.9888545).
- [39] D. Oviedo, M. C. Romero-Ternero, M. D. Hernández, F. Sivianes, A. Carrasco, and J. I. Escudero, "Multiple intelligences in a MultiAgent system applied to telecontrol," *Exp. Syst. Appl.*, vol. 41, no. 15, pp. 6688–6700, Nov. 2014.
- [40] S. Tasdemir, B. Yaniktepe, and A. B. Guher, "The effect on the wind power performance of different normalization methods by using multilayer feed-forward backpropagation neural network," *Int. J. Energy Appl. Technol.*, vol. 5, no. 3, pp. 131–139, Dec. 2018, doi: [10.31593/ijeat.464210](https://doi.org/10.31593/ijeat.464210).
- [41] A. Haris, S. Wahjuni, H. Sukoco, H. Rahmawan, S. N. Neyman, H. Sikumbang, and M. J. Elly, "Technology sun tracking system for solar power plants base on recurrent neural networks," in *Proc. Int. Seminar Sci. Appl. Technol. (ISSAT)*. Bandung, Indonesia: Atlantis Press, 2020, pp. 223–226, doi: [10.2991/aer.k.201221.038](https://doi.org/10.2991/aer.k.201221.038).
- [42] J. D. Kaul, G. D. Weed, J. Cunningham, I. Ahmed, and W.-S. Yoo, "Prototype development for adaptive solar tracking and optimization of data communication protocol," in *Proc. ASEE North Central Sect. Conf.*, 2021, p. 8.
- [43] B. G. Pierce, J. L. Braid, J. S. Stein, J. Augustyn, and D. Riley, "Solar transposition modeling via deep neural networks with sky images," *IEEE J. Photovolt.*, vol. 12, no. 1, pp. 145–151, Jan. 2022, doi: [10.1109/JPHOTOV.2021.3120508](https://doi.org/10.1109/JPHOTOV.2021.3120508).
- [44] C. Tian, J. Ma, C. Zhang, and P. Zhan, "A deep neural network model for short-term load forecast based on long short-term memory network and convolutional neural network," *Energies*, vol. 11, no. 12, p. 3493, Dec. 2018, doi: [10.3390/en11123493](https://doi.org/10.3390/en11123493).
- [45] E. F. Syahram and M. M. Effendy, "Sun position forecasting using the RNN method—LSTM as a solar cell power control reference," *J. Electr. Eng. Technol.*, vol. 3, no. 1, p. 8, No. 2021.
- [46] S. F. Stefenon, C. Kasburg, A. Nied, A. C. R. Klaar, F. C. S. Ferreira, and N. W. Branco, "Hybrid deep learning for power generation forecasting in active solar trackers," *IET Gener., Transmiss. Distribution*, vol. 14, no. 23, pp. 5667–5674, Dec. 2020, doi: [10.1049/iet-gtd.2020.0814](https://doi.org/10.1049/iet-gtd.2020.0814).
- [47] W. Lee, K. Kim, J. Park, J. Kim, and Y. Kim, "Forecasting solar power using long-short term memory and convolutional neural networks," *IEEE Access*, vol. 6, pp. 73068–73080, 2018, doi: [10.1109/ACCESS.2018.2883330](https://doi.org/10.1109/ACCESS.2018.2883330).
- [48] A. Géron, *Hands-on Machine Learning With Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*, 2nd ed. Sebastopol, CA, USA: O'Reilly Media, 2019.
- [49] E. Bisong, "Google colaboyatory," in *Building Machine Learning and Deep Learning Models on Google Cloud Platform*. Berkeley, CA, USA: Apress, 2019, pp. 59–64, doi: [10.1007/978-1-4842-4470-8_7](https://doi.org/10.1007/978-1-4842-4470-8_7).
- [50] S.-O. Park, O. Kwon, Y. Kim, S. K. Cha, and H. Yoon, "Mind control attack: Undermining deep learning with GPU memory exploitation," *Comput. Secur.*, vol. 102, Mar. 2021, Art. no. 102115, doi: [10.1016/j.cose.2020.102115](https://doi.org/10.1016/j.cose.2020.102115).
- [51] G. Terrén-Serrano, A. Bashir, T. Estrada, and M. Martínez-Ramón, "Girasol, a sky imaging and global solar irradiance dataset," *Dryad*, 2020, Art. no. 118473724459, doi: [10.5061/DRYAD.ZCRJDFN9M](https://doi.org/10.5061/DRYAD.ZCRJDFN9M).
- [52] G. Terrén-Serrano, A. Bashir, T. Estrada, and M. Martínez-Ramón, "Girasol, a sky imaging and global solar irradiance dataset," *Data Brief*, vol. 35, Apr. 2021, Art. no. 106914, doi: [10.1016/j.dib.2021.106914](https://doi.org/10.1016/j.dib.2021.106914).
- [53] J. Sillmann, T. Thorarindottir, N. Keenlyside, N. Schaller, L. V. Alexander, G. Hegerl, S. I. Seneviratne, R. Vautard, X. Zhang, and F. W. Zwiers, "Understanding, modeling and predicting weather and climate extremes: Challenges and opportunities," *Weather Climate Extremes*, vol. 18, pp. 65–74, Dec. 2017, doi: [10.1016/j.wace.2017.10.003](https://doi.org/10.1016/j.wace.2017.10.003).
- [54] Y. Zhang, Y. Zhang, and A. Haghani, "A hybrid short-term traffic flow forecasting method based on spectral analysis and statistical volatility model," *Transp. Res. C, Emerg. Technol.*, vol. 43, pp. 65–78, Jun. 2014, doi: [10.1016/j.trc.2013.11.011](https://doi.org/10.1016/j.trc.2013.11.011).
- [55] T. Ouyang, A. Kusiak, and Y. He, "Predictive model of yaw error in a wind turbine," *Energy*, vol. 123, pp. 119–130, Mar. 2017, doi: [10.1016/j.energy.2017.01.150](https://doi.org/10.1016/j.energy.2017.01.150).
- [56] S. Bhanja and A. Das, "Impact of data normalization on deep neural network for time series forecasting," 2018, *arXiv:1812.05519*.
- [57] G. Aksu, C. O. Güzeller, and M. T. Eser, "The effect of the normalization method used in different sample sizes on the success of artificial neural network model," *Int. J. Assessment Tools Educ.*, vol. 6, no. 2, pp. 170–192, Jul. 2019, doi: [10.21449/ijate.479404](https://doi.org/10.21449/ijate.479404).
- [58] A. Akilli and H. Atil, "Evaluation of normalization techniques on neural networks for the prediction of 305-day milk yield," *Turkish J. Agricult. Eng. Res.*, pp. 354–367, Dec. 2020, doi: [10.46592/turkager.2020.v01i02.011](https://doi.org/10.46592/turkager.2020.v01i02.011).
- [59] Z. Wang and T. Oates, "Imaging time-series to improve classification and imputation," 2015, *arXiv:1506.00327*.

- [60] C.-L. Yang, C.-Y. Yang, Z.-X. Chen, and N.-W. Lo, "Multivariate time series data transformation for convolutional neural network," in *Proc. IEEE/SICE Int. Symp. Syst. Integr. (SII)*. Paris, France: IEEE, Jan. 2019, pp. 188–192, doi: [10.1109/SII.2019.8700425](https://doi.org/10.1109/SII.2019.8700425).
- [61] S. Oehmcke, O. Zielinski, and O. Kramer, "Input quality aware convolutional LSTM networks for virtual marine sensors," *Neurocomputing*, vol. 275, pp. 2603–2615, Jan. 2018, doi: [10.1016/j.neucom.2017.11.027](https://doi.org/10.1016/j.neucom.2017.11.027).
- [62] S. Indolia, A. K. Goswami, S. P. Mishra, and P. Asopa, "Conceptual understanding of convolutional neural network—A deep learning approach," *Proc. Comput. Sci.*, vol. 132, pp. 679–688, 2018, doi: [10.1016/j.procs.2018.05.069](https://doi.org/10.1016/j.procs.2018.05.069).
- [63] H. Z. Wang, G. B. Wang, G. Q. Li, J. C. Peng, and Y. T. Liu, "Deep belief network based deterministic and probabilistic wind speed forecasting approach," *Appl. Energy*, vol. 182, pp. 80–93, Nov. 2016, doi: [10.1016/j.apenergy.2016.08.108](https://doi.org/10.1016/j.apenergy.2016.08.108).
- [64] Y. Chen, S. Zhang, W. Zhang, J. Peng, and Y. Cai, "Multifactor spatio-temporal correlation model based on a combination of convolutional neural network and long short-term memory neural network for wind speed forecasting," *Energy Convers. Manage.*, vol. 185, pp. 783–799, Apr. 2019, doi: [10.1016/j.enconman.2019.02.018](https://doi.org/10.1016/j.enconman.2019.02.018).
- [65] T. Beysolow II, "Introduction to deep learning," in *Introduction to Deep Learning Using R*. Berkeley, CA, USA: Apress, 2017, pp. 1–9, doi: [10.1007/978-1-4842-2734-3_1](https://doi.org/10.1007/978-1-4842-2734-3_1).
- [66] E. Ahmadzadeh, H. Kim, O. Jeong, N. Kim, and I. Moon, "A deep bidirectional LSTM-GRU network model for automated ciphertext classification," *IEEE Access*, vol. 10, pp. 3228–3237, 2022, doi: [10.1109/ACCESS.2022.3140342](https://doi.org/10.1109/ACCESS.2022.3140342).
- [67] S. Gao, Y. Huang, S. Zhang, J. Han, G. Wang, M. Zhang, and Q. Lin, "Short-term runoff prediction with GRU and LSTM networks without requiring time step optimization during sample generation," *J. Hydrol.*, vol. 589, Oct. 2020, Art. no. 125188, doi: [10.1016/j.jhydrol.2020.125188](https://doi.org/10.1016/j.jhydrol.2020.125188).
- [68] J. Wang, W. Yang, P. Du, and T. Niu, "A novel hybrid forecasting system of wind speed based on a newly developed multi-objective sine cosine algorithm," *Energy Convers. Manage.*, vol. 163, pp. 134–150, May 2018, doi: [10.1016/j.enconman.2018.02.012](https://doi.org/10.1016/j.enconman.2018.02.012).
- [69] Y. Xu, W. Yang, and J. Wang, "Air quality early-warning system for cities in China," *Atmos. Environ.*, vol. 148, pp. 239–257, Jan. 2017, doi: [10.1016/j.atmosenv.2016.10.046](https://doi.org/10.1016/j.atmosenv.2016.10.046).
- [70] T. O. Hodson, T. M. Over, and S. S. Foks, "Mean squared error, deconstructed," *J. Adv. Model. Earth Syst.*, vol. 13, no. 12, Dec. 2021, Art. no. e2021MS002681, doi: [10.1029/2021ms002681](https://doi.org/10.1029/2021ms002681).
- [71] D. S. K. Karunasingha, "Root mean square error or mean absolute error? Use their ratio as well," *Inf. Sci.*, vol. 585, pp. 609–629, Mar. 2022, doi: [10.1016/j.ins.2021.11.036](https://doi.org/10.1016/j.ins.2021.11.036).
- [72] C. Chen, J. Twycross, and J. M. Garibaldi, "A new accuracy measure based on bounded relative error for time series forecasting," *PLoS ONE*, vol. 12, no. 3, Mar. 2017, Art. no. e0174202, doi: [10.1371/journal.pone.0174202](https://doi.org/10.1371/journal.pone.0174202).
- [73] T. Chai and R. R. Draxler, "Root mean square error (RMSE) or mean absolute error (MAE)?—Arguments against avoiding RMSE in the literature," *Geoscientific Model Develop.*, vol. 7, no. 3, pp. 1247–1250, Jun. 2014, doi: [10.5194/gmd-7-1247-2014](https://doi.org/10.5194/gmd-7-1247-2014).
- [74] E. González-Estrada and W. Cosmes, "Shapiro-wilk test for skew normal distributions based on data transformations," *J. Stat. Comput. Simul.*, vol. 89, no. 17, pp. 3258–3272, Nov. 2019, doi: [10.1080/00949655.2019.1658763](https://doi.org/10.1080/00949655.2019.1658763).
- [75] W. U. H. Shah, G. Hao, H. Yan, N. Zhu, R. Yasmeen, and G. Dincă, "Role of renewable, non-renewable energy consumption and carbon emission in energy efficiency and productivity change: Evidence from G20 economies," *Geosci. Frontiers*, May 2023, Art. no. 101631, doi: [10.1016/j.gsf.2023.101631](https://doi.org/10.1016/j.gsf.2023.101631).



MWENGE MULENGA received the B.Sc. and M.Sc. degrees in computer systems engineering from Saint Petersburg Electrotechnical University, Saint Petersburg, Russia, in 2007 and 2009, respectively, and the Ph.D. degree in computer science from the University of Malaya, Malaysia, in 2022. He specializes in software engineering, artificial intelligence, and data science. He was a Lecturer in computer science with the School of Engineering and Technology, Mulungushi University, Kabwe, Zambia, where he was also the Assistant Dean, from 2011 to 2018. He is currently a Senior Lecturer with the National Institute of Public Administration, Lusaka, Zambia. His research interests include disease identification using microbiome data and images and the application of learning models in solar tracking systems.



MUSA PHIRI received the B.Sc. degree in statistics from Mulungushi University, Kabwe, Zambia, in 2016, the M.Sc. degree in project management from the University of Zambia, Lusaka, Zambia in 2020, and the M.Sc. degree in data science and big data from Mulungushi University, in 2023. His research interests include the intersection of artificial intelligence and renewable energy, particularly in exploring the applications of deep learning in solar tracking systems and solar photovoltaic power generation. In recent years, he has collaborated actively with researchers in several other domains, such as natural language processing and health.



LUCKSON SIMUKONDA received the B.Sc. degree in computer science from Mulungushi University, Kabwe, Zambia, in 2015, and the M.Sc. degree in control engineering and robotics from the Gdańsk University of Technology, Gdańsk, Poland, in 2019. He is currently a Lecturer and a Researcher with the School of Engineering and Technology, Mulungushi University. His research interests include automatic control systems, robotics, computer vision, embedded systems, and artificial intelligence.



FADELE A. ALABA received the Ph.D. degree in computer science from the University of Malaya, Malaysia. He is currently a Researcher with the Department of Computer Science, Federal College of Education, Zaria, Kaduna. His current research interests include network security, IT security, and cloud computing.

• • •