



Review Article

Deep learning algorithms for very short term solar irradiance forecasting: A survey

Meenu Ajith^{a,*}, Manel Martínez-Ramón^b^a Tri-Institutional Center for Translational Research in Neuroimaging and Data Science, Georgia State University, Georgia Institute of Technology, Emory University, Atlanta, 30303, GA, USA^b Electrical and Computer Engineering Department, The University of New Mexico, Albuquerque, 87106, NM, USA

ARTICLE INFO

Keywords:

Solar irradiance forecasting
Multimodal deep learning
Hybrid models
Long short term memory
Optical flow
Convolutional neural networks
Infra-red images

ABSTRACT

Integrating solar energy with existing grid systems is difficult due to its variability, which is impacted by factors such as the predicted horizon, meteorological conditions, and geographic position. Accurate global horizontal irradiance (GHI) estimates can help to address this issue and allow for early and effective participation in the energy planning and management market. The existing models either use time series data or sky images in various network topologies to perform solar radiance forecasts. This study compares three categories of solar irradiance forecasting models such as time series-based, image-based and hybrid models. Here several state-of-the-art methods are compared against the proposed models, namely Convolutional Long Short-Term Memory Fusion Network (CNN-L) and Multiple Image Convolutional Long Short-Term Memory Fusion Network (MICNN-L). Both models use both infrared sky images as well as past values of GHI for prediction. These methods extract spatial features using convolutional neural networks and temporal features using long short-term memory networks. The extracted features are finally concatenated and passed through a fully connected layer to obtain a prediction. Further analysis also included using a feature extraction method such as optical flow (OF) on the image data before passing it to the hybrid model MICNN-L (OF). The results observed in this comparative analysis denote that MICNN-L improves the efficacy of the forecasts in cloudy conditions compared to the rest of the state-of-the-art approaches.

1. Introduction

Global energy demand has increased tremendously in recent years due to the depletion of non-renewable energy sources such as fossil fuels and nuclear power. Consequently, various alternative sources got integrated into power grids for large-scale energy production. Amongst the available sources, the most reliable one is solar energy since it is one of the inexhaustible renewable energy sources [1]. However, the feasibility of these systems depends highly on climatic conditions such as the cloudiness of the sky, humidity, and temperature, which change rapidly over a short duration. Hence, due to the intermittent nature of solar power, precise solar radiation prediction is required for the systematic functioning of the power grids.

Solar irradiance data is typically predicted for varying forecasting horizons depending on the specific application in question. Very short-term solar irradiance forecasting involves predicting irradiance values over a short duration, typically ranging from a few seconds to a few minutes ahead. This helps manage the fluctuations in energy supply due to the rapidly changing weather conditions and supports real-time

decision-making. Additionally, accurate and timely solar irradiance predictions on a very short-term basis can help solar energy systems adjust their operations to maintain a stable energy output, ensuring a consistent grid supply to the grid [2]. The methods that are effective for very short-term solar irradiance forecasting may not necessarily be as effective for long-term forecasting [3]. This is because the factors that affect solar irradiance vary over different time scales. Very short-term solar irradiance forecasting is influenced by rapidly changing atmospheric conditions whereas long-term forecasting may require consideration of more slow-changing factors, such as seasonal variations, climate patterns, and long-term trends [4]. The present study endeavors to establish a comprehensive taxonomy of the methods employed for very short-term forecasting of solar irradiance.

A range of forecasting methods, including persistence model [5], physical models, satellite models, statistical models, and machine learning-based models, can be utilized for solar irradiance prediction. The physical models [6] include the numerical weather prediction (NWP) methods, mainly used for predicting longer time horizons such

* Corresponding author.

E-mail addresses: majith@gsu.edu (M. Ajith), manel@unm.edu (M. Martínez-Ramón).<https://doi.org/10.1016/j.rser.2023.113362>

Received 16 November 2022; Received in revised form 8 May 2023; Accepted 10 May 2023

Available online 29 May 2023

1364-0321/© 2023 Elsevier Ltd. All rights reserved.

Abbreviations

Adam	Adaptive Moment Optimization
CNN	Convolutional Neural Networks
CNN-L	Convolutional Long Short Term Memory Fusion Network
DNI	Direct Normal Irradiance
ESN	Echo State Network
FS	Forecasting Skill
GBC	Ground-Based Cloud Images
GHI	Global Horizontal Irradiance
GP	Gaussian Processes
LSTM	Long Short-Term Memory
MAPE	Mean Absolute Percentage Error
MICNN-L	Multiple Image Convolutional Long Short Term Memory Fusion Network
MICNN-L(OF)	Multiple Image Convolutional Long Short Term Memory Fusion Network Optical Flow
MLP	Multilayer Perceptron
nMAE	Normalized Mean Absolute Error
nMBE	Normalized Mean Bias Error
NREL	National Renewable Energy Laboratory
nRMSE	Normalized Root Mean Square Error
r	Correlation Coefficient
ReLU	Rectified Linear Unit
RNN	Recurrent Neural Network
SCNN-LSTM	Siamese Convolutional Neural Network-Long Short-Term Memory
SGD	Stochastic Gradient Descent
SRRL	Solar Radiation Research Laboratory
SVM	Support Vector Machines
TSI	Total Sky Images

Nomenclature

α_n^*	Lagrange multiplier used for constraints defined in negative errors
α_n	Lagrange multiplier used for constraints defined in positive errors
b	Vector of biases used in an estimator
\mathbf{c}_k	Vector of functions used as inner state gate in step k of an LSTM or GRU network
\mathbf{k}	Vector of functions used as a forget gate in step k of an LSTM or GRU network
\mathbf{i}_k	Vector of functions used as input gate in step k of an LSTM or GRU network
K	Kernel matrix which contain dot products between vectors
$\mathbf{k}(\cdot)$	Vector of kernel dot products used in kernel-based methods
\mathbf{o}_k	Vector of functions used as output in step k of a RNN, LSTM or GRU network
w	Column vector containing the parameters of a linear estimator
\mathbf{x}_k	Column vector consisting of a sequence of samples x_k
z	Column vectors of estimated outputs
\mathcal{H}	Reproducing kernel Hilbert space function
$\sigma(\cdot)$	Activation function in a neural network
σ_n^2	Variance of the Gaussian model
$\tilde{\mathbf{c}}_k$	Vector of functions used as output state gate in step k of an LSTM network

$\varphi(\cdot)$	Nonlinear mapping function that maps data into a Hilbert space
C	Trade-off parameter used to balance the structural and the empirical risks in an SVM
e_k	Error estimate at time instant k
$f(\cdot)$	An estimation function used to approximate a system
k	Number of samples
$k(\cdot, \cdot)$	Mercer's kernel that measures the similarity between 2 inputs
L	Number of nodes
T	Duration of the time series
x_t	A time series sample

as day-ahead forecasting. Satellite-based models utilize weather satellite data at 10 to 15-min intervals to forecast solar irradiance over time scales ranging from 1 to 6 h, covering large geographic areas. [7]. These models may not be well-suited for very short-term forecasting of solar irradiance because of their limited temporal resolution. In contrast, persistence models [8] assume that solar irradiance will remain constant over a short period of time. Due to its assumption of unchanging atmospheric conditions, this model may not perform effectively in the presence of unprecedented or rapidly changing weather conditions. Therefore, to improve the accuracy of solar irradiance predictions, various statistical models such as autoregressive moving integrated average (ARIMA) [9], exponential smoothing and regression models were used to create a predictive model from the available historical data. Traditional statistical methods have been challenged by the nonlinearity and non-stationarity of solar radiation, leading to the growing dominance of machine learning-based approaches [10]. Such methods have been found to be more efficient in predicting the dynamic relationships between solar radiation and its related factors. Hence, the shift towards machine learning algorithms represents a promising approach to modeling and predicting solar radiation in a more accurate and comprehensive manner.

This research study presents a comprehensive and comparative evaluation of various methodologies for predicting solar irradiance based on input. The input can take the form of single or multidimensional arrays, including time series, images, or a combination of both. The time series methods employed in this study are developed using linear models, kernel learning, feed-forward networks, and recurrent networks. Conversely, models that use only images are analyzed using single or multidimensional convolutional neural networks. The hybrid models were also developed and analyzed by combining the methods used in time series and image-only models. Therefore, the purpose of this comparative study is to systematically address several research gaps in the field of very short-term solar irradiance forecasting by incorporating all significant methods for this problem. This aims to provide valuable insights into the strengths and weaknesses of each method, as well as their suitability for different types of input data, ultimately contributing to the advancement of solar energy forecasting and utilization. The main contributions of this paper are as follows:

- As far as our knowledge extends, no literature has comprehensively studied the use of infrared-based hybrid models for very short-term solar irradiance forecasting. In comparison to ground-based cloud images (GBC) and total sky images (TSI), infrared images [11] have the advantage of being able to detect temperature differences. This helps to provide additional information about cloud formation and movement. This information can be particularly useful in predicting changes in solar irradiance due to cloud cover [12]. Additionally, infrared images can provide more reliable data in situations where visible light may be obstructed, such as during nighttime or under heavy cloud cover.

- Two models namely Convolutional Long Short-Term Memory Fusion Network (CNN-L) and Multiple Image Convolutional Long Short-Term Memory Fusion Network (MICNN-L), were developed previously by the authors for solar irradiance forecasting for a few seconds ahead [13]. This work extended the work for 10 min ahead of prediction and introduced a new variation namely MICNN-L(OF). This model uses optical flow-based feature extraction on the consecutive infra-red images before passing it into the MICNN-L model to further aid the prediction process. The MICNN-L models show better performance compared to the rest of the models.
- A comparative analysis that highlights the advantages and shortcomings of very short-term solar irradiance forecasting models. The study focuses on the trade-off between model complexity and performance. Hybrid models, while more accurate, have a greater number of trainable parameters and higher complexity. In contrast, time series models use fewer parameters and are less accurate. Based on the specific requirements of the problem at hand, an appropriate model can be selected.

The rest of the paper is organized as follows. Section 2 presents related works that include the different types of machine-learning models used for very short-term solar irradiance forecasting. The materials and methods employed for the comparative study are described in Section 3. Section 4 provides a detailed analysis of the experimental design and training criteria of all the methods. Section 5 demonstrates a discussion highlighting the complexity and performance of the different models mentioned in the study. Finally, Section 6 provides the conclusion and prospects of the present study.

2. Related works

The machine learning methods used for solar irradiance forecasting can be broadly categorized into three groups based on the input data used by the models: time series, image-based, and hybrid models. For time series models, support vector machines (SVMs) [14], artificial neural networks (ANNs) [15], and Gaussian processes (GPs) [16] have demonstrated superior performance compared to linear regression. To evaluate the efficacy of these supervised approaches, a clearness index was computed based on the correlation between weather conditions and incident solar radiation [17]. However, these models were insufficient in resolving the challenge of power fluctuations experienced during days with cloudy weather. As a result, recurrent neural networks (RNNs) [18] and echo state networks (ESN) [19] were utilized for time series forecasting of solar irradiance. However, during the training process of an RNN, a challenge known as exploding gradients [20–22] emerged, whereby the norm of gradients increases and impedes convergence during training. Hence this network can be difficult to train and require more time steps from the past to yield accurate results for prediction. On the other hand, ESNs cannot capture complex temporal dependencies due to their fixed and random internal reservoir weights. In contrast, LSTM networks have introduced gating mechanisms that facilitate a more efficient flow of information within the network [23]. This has enabled them to efficiently model long-range dependencies for better prediction [24]. A study showed that LSTMs achieved an 18.34% increase in accuracy compared to ANNs for very short-term forecasting [25]. After LSTMs, Gated Recurrent Units (GRUs) were introduced as a computationally efficient variant of RNNs. Although both models demonstrate comparable performance, GRUs have fewer parameters, resulting in faster training time. In research comparing the forecasting performance of LSTM and GRU for very short-term predictions, LSTMs showed a slightly lower prediction error for a 10-min ahead forecast [26]. However, for longer time horizons of 15 and 20 min, GRU outperformed LSTM.

Image-based models for solar irradiance forecasting usually include convolutional neural networks (CNNs) [27]. The 1D-CNN models are

also used as time series prediction models as they can train the network much faster than recurrent networks that process data sequentially [28]. With the advent of various technologies related to digital image processing [29] in recent years, different instruments for cloud-measuring, such as the total sky imager, have been developed successfully. They can monitor and gather cloud images simultaneously from photovoltaic power stations [30], allowing the researchers to correlate these images with solar irradiance. Thus, the CNNs were used along with sky images as input for predicting the PV output. Later, the SolarNet model was developed to automatically extract the characteristics from the total sky image [31]. However, this network solely relied on a single total sky image and thus failed to account for cloud motion information, which significantly impacted the accuracy of the forecasts.

In recent years, hybrid models have become the most popular and widely developed models for solar irradiance forecasting. In one study, a hybrid model for solar irradiance forecasting utilized a 3D-CNN architecture that integrated both images and historical values [32]. The fused features generated from this model were then fed into a multi-layer perceptron (MLP) to produce the final predictions. However, since the MLP cannot capture the long-term dependencies between the input features, it results in poor performance during the forecast. Considering the limitations of the 3D-CNN model, a new hybrid approach, namely the Siamese Convolutional Neural Network-Long Short-Term Memory (SCNN-LSTM) network, was developed for solar irradiance forecasting [33]. A Siamese CNN was able to extract both spatial and temporal information from consecutive total sky images. They were concatenated with meteorological information to fuse image characteristics and the resulting features were fed into an LSTM model to estimate solar irradiance. The total sky images could capture the full sky dome and the distribution of clouds in all directions, including cloud height and thickness. However, this equipment was expensive, and the cloud mapping process requires significant computational resources. Recently, another approach for solar irradiance forecasting was proposed based on ensemble learning [34]. This method considers both GBC images and meteorological variables as fused inputs to the model. The multimodal ensemble model had poorer performance compared to the 3D-CNN due to its increased complexity and lack of generalization.

3. Materials and methods

The focus of this study was to compare and evaluate the effectiveness of different machine learning models for very short-term solar irradiance forecasting. This section includes a comprehensive summary of the dataset used by each model, the preprocessing techniques applied to the data, and the basic structure and theoretical background of each method. The machine learning and deep learning methods used in this analysis are represented in Fig. 1. The models used in this study consist of SVM, GP, MLP, LSTM, RNN, GRU, and 1D-CNN, all of which were trained exclusively on past solar irradiance data. On the other hand, the 2D-CNN and SolarNet models were trained using image data, setting them apart from the other models in terms of data type. Additionally, the study explored the performance of several hybrid models, namely 3D-CNN, SCNN-LSTM, CNN-L, MICNN-L, and MICNN-L(OF), to evaluate their capability of combining time series and image data for forecasting. Table 1 shows a detailed description of the type of input data and preprocessing steps used by each of the methods.

3.1. Area of study and data collection

This study includes various methods that use past values of solar irradiance, GBC images, TSIs, and IR images as their input. The past values of solar irradiance and IR images were gathered by employing a data acquisition system (DAQ) consisting of a camera and a pyranometer. This model is positioned on the roofing of the building in the Mechanical Engineering department. It is further located at the

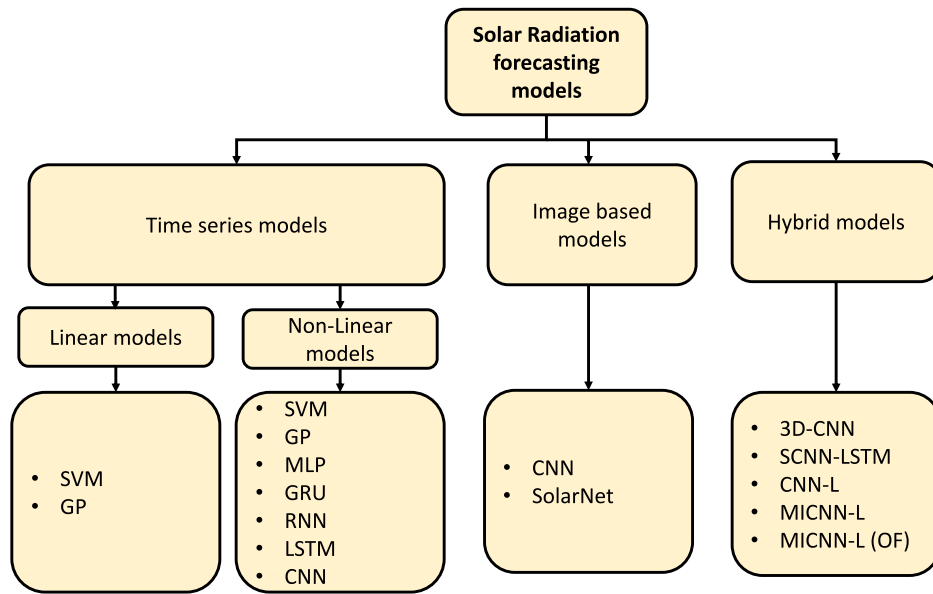


Fig. 1. Different categories of models for solar irradiance forecasting.

University of New Mexico, situated in the middle of Albuquerque, New Mexico, USA. This city experiences a mild semi-arid climate with chilly winters, hot summers, and occasional showers that are more likely to fall during the summer. The city is situated on a plateau around 35 degrees north latitude. The eastern districts, which are at the foot of the Sandia Mountains, rise to 1900 m, while the center is at 1500/1600 m (4900/5200 ft) (6200 ft). The sky is either clear or slightly overcast over 80 percent during mid-May and mid-June. On average, 170 days a year are bright with fewer than 30% of clouds present, and 110 days are slightly sunny with 40%–80% of clouds. The temperature ranges from a low of -4°C in the winter to a high of roughly 33°C in summer. Additionally, about 11 inches of rain and snow accumulate annually.

The infrared images and time-series information for solar irradiance forecasting are obtained simultaneously when one of the systems collects the circumsolar IR images. On the other hand, the pyranometer is utilized to estimate the GHI values. Additionally, the DAQ contains a solar tracker that continuously pans and tilts, maintaining the sun's center line throughout the day. A FLIR Lepton[®] radiometric camera with an 8–14 mm wavelength serves as the IR sensor that results in a constant thermal image [35]. A frame's pixels are measurements of emission temperature in centi-kelvin, denoted as temperature. A set of 10 consecutive images is captured every 15 s, averaged to lessen noise, and saved in the png configuration. An analog antialiasing filter with a sampling frequency of 4–6 samples/s handles the pyranometer output. Hence, before training, the data undergoes noise removal and antialiasing filtering. Finally, the final data format is a single sample every 15 s. This data is used as input in SVM, GP, MLP, RNN, GRU, LSTM, CNN, CNN-L, MICNN-L, and MICNN-L (OF).

Models like SolarNet, 3D-CNN, and SCNN-LSTM that utilize TSIs and GBC images as input relies on the NREL solar radiation research laboratory (SRRL) dataset [36] as their source of data. Since 1981, the SRRL has been continuously collecting solar measurements at the South Table Mountain Campus of NREL in Golden, Colorado. The SRRL station is situated at an elevation of 1828.2 m, at latitude 39.74°N and longitude 105.18°W . This location is renowned for its abundant solar resources. The SRRL measures various meteorological variables, such as GHI, direct normal irradiance, direct horizontal irradiance, temperature, relative humidity, wind speed, and atmospheric pressure, with a 1-min sampling frequency. Additionally, the SRRL dataset includes two sets of total TSIs, captured at a 10-min interval, using a Yankee Total Sky Imager (TSI-800) and an EKO All Sky Imager, respectively.

3.2. Time series models

The time-series data defined as x_t consists of k samples and the predictions are made for the time horizon T . The correlation between the past and future sample x_{k+T} can be used to define a parametric function as follows:

$$x_{k+T} = f(\mathbf{x}_k) + e_k \quad (1)$$

where \mathbf{x}_k is a column vector consisting of a frame of L previous values of given time-series and e_k is the error during prediction. Here the time series data refers to the GHI data recorded from the pyranometer. The linear and non-linear time series models use only this information to predict the values at the future time horizon. The most commonly used time series models for solar irradiance forecasting are discussed below:

3.2.1. Support Vector Machines (SVM)

The SVM approach [40–44] is a learning machine optimized by the so-called maximum margin criterion [45] that is expressed in a kernel reproducing Hilbert space [46] thus having nonlinear properties. Namely, such a machine can be formulated as

$$f(\mathbf{x}) = \mathbf{w}^T \varphi(\mathbf{x}) + b \quad (2)$$

where function $\varphi(\cdot)$ transforms the input data \mathbf{x} into a Hilbert space \mathcal{H} provided with a dot product that can be represented as a function of the input patterns as $\langle \mathbf{x}, \mathbf{x}' \rangle = k(\mathbf{x}, \mathbf{x}')$, $\mathbf{w} \in \mathcal{H}$ is a vector of parameters to be optimized and b is a bias term. For such a function, usually called a Mercer's kernel, to be a dot product, it only needs to be a positive definite function, according to Mercer's theorem [47]. Given a set of N training samples $\mathbf{x}_1, \dots, \mathbf{x}_N$ and by virtue of the Generalized Representer Theorem [48], these machines can be expressed, under nonrestrictive conditions, as $f(\mathbf{x}) = \sum_{i=1}^N \alpha_i k(\mathbf{x}_i, \mathbf{x}) + b$, where \mathbf{w} is substituted by its dual equivalent $\mathbf{w} = \sum_{i=1}^N \alpha_i \varphi(\mathbf{x}_i)$.

The particular algorithm of SVM for regression was introduced in [49]. In these machines, a cost function is established that weights to zero these estimation errors whose absolute value is less than a given quantity ϵ , and assigns a cost equal to $C(\|e\| - \epsilon)$ if $\|e\| \geq \epsilon$. The resulting machine is formulated as

$$f(\mathbf{x}) = \sum_{i=1}^N (\alpha_i - \alpha_i^*) k(\mathbf{x}_i, \mathbf{x}) + b \quad (3)$$

A constrained optimization gives the solution for the coefficients $\alpha_i - \alpha_i^*$, which is the derivative of the cost function with respect to the error.

Table 1
Detailed analysis of the input data and preprocessing of all the discussed methods.

Input data modality	Input data	Data preprocessing	Method
Time series	Past values of pyranometric values of solar irradiance	Antialiasing filtering and subsampling	SVM (Linear/Nonlinear) [14] GP (Linear/Nonlinear) [16] MLP [15,37] GRU [26] RNN [18] LSTM [23–25,37,38] 1D-CNN [28]
	Single infrared image	Image averaging and normalization	2D-CNN [28]
Image	Single total sky image	Image averaging, normalization and binary masking	SolarNet [31]
	A sequence of ground-based images and past values of solar irradiance	Time series: None Image: Binary masking	3D-CNN [39]
Image, time series	A sequence of total sky images, past values of solar irradiance and meteorological observations	Time series: Converting to clear sky index Image: Subsampling and binary masking	SCNN-LSTM [33]
	Single infra-red image and past values of solar irradiance	Time series: Antialiasing filtering and subsampling	CNN-L [37]
	A sequence of infra-red images and past values of solar irradiance	Image: Averaging and normalization	MICNN-L [13]
	A sequence of infra-red images that underwent optical flow and past values of solar irradiance		MICNN-L (OF) [13]

This is, they are zero for those samples that produce an estimation error less than ϵ . If the error is positive and higher than ϵ , then $\alpha_i = C$ and $\alpha_i^* = 0$. In case of negative errors, $\alpha_i = 0$, and $\alpha_i^* = C$. This machine is then sparse and its result is, when free parameters C and ϵ are adequate, robust against outliers. The drawback of this methodology is that it requires the cross-validation of these parametric values, usually with a validation data set that must be set aside for this purpose. The chosen Mercer's kernel function is also parametric and its parameters need also to be validated. This approach is particularly well suited when the number of training samples is limited, due to the strong regularization mechanism of the SVM. On the other side, this machine may pose computational difficulties when the number of samples is high since the computational burden of the optimization algorithm is $\mathcal{O}(N^3)$. Multitask SVM approaches [50] exist that can be used in multi-horizon solar forecast [51].

3.2.2. Gaussian Processes (GP)

The GP approach, as the SVM, can be classified among the Kernel methods, this is, the structure of the estimation function is formally the same, but the optimization criterion is based on maximum likelihood. The GP model [52] can be represented as an estimator using the below form.

$$y = \mathbf{w}^\top \varphi(\mathbf{x}) + e \quad (4)$$

where y is the regressor to be predicted and e is the prediction error, which is modeled as a sequence of independent and identically distributed zero-mean Gaussian with variance σ_n^2 . In this equation, the bias b is included inside the set of parameters \mathbf{w} for convenience. The Gaussian model for the noise leads, together with the independence assumption, to a straightforward Gaussian model for the likelihood of the sequence of training regressors y_i as a function of the parameters and the training observations \mathbf{x}_i . Then, a prior distribution $p(\mathbf{w})$ for the primal parameters \mathbf{w} is stated as a multivariate Standard. This, combined with the likelihood $p(\mathbf{y}|\mathbf{X}, \mathbf{w})$ (where \mathbf{y} and \mathbf{X} are the column vector and matrix of training regressors and inputs respectively) and by utilizing the Bayes rule, a posterior distribution is formulated from the primal parameters, which is also a Gaussian distribution. With this result, a Gaussian posterior distribution is found for a test sample \mathbf{x}^* , with mean and variance

$$\begin{aligned} \tilde{f}(\mathbf{x}^*) &= \mathbf{y}^\top (\mathbf{K} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{k}(\mathbf{x}^*) \\ \sigma_*^2 &= k(\mathbf{x}^*, \mathbf{x}^*) - \mathbf{k}^\top(\mathbf{x}^*) (\mathbf{K} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{k}(\mathbf{x}^*) \end{aligned} \quad (5)$$

where \mathbf{K} denotes the kernel matrix containing the dot products between training predictors $k(\mathbf{x}_i, \mathbf{x}_j)$. The column vector $\mathbf{k}(\mathbf{x}^*)$ contains the dot products $K(\mathbf{x}^*, \mathbf{x}_j)$ between the training and the test sample, and the identity matrix is represented as \mathbf{I} .

The advantage of this approach over the SVM is that here the optimization of all the parameters is done by maximizing the marginal log-likelihood of the regressors during training. This is commonly done during gradient descent, therefore avoiding the necessity of cross-validating them with a separate validation data set. The use of a Bayesian approach provides the method with automatic regularization capabilities, and therefore this method is robust in problems cases where the number of data is limited. The fact that the GP models the prediction with a posterior probability adds the possibility of computing the estimated interval of confidence of the prediction, which allows the user to estimate the validity or the quality of the prediction. As in the SVM, the method may pose computational problems due to the fact that the computation of the solution has a computational burden of $\mathcal{O}(N^3)$. The main trade-off of this methodology is that a probabilistic model is needed for the noise and the result may be far away from being optimal if this model is not accurate. Particularly interesting in the solar forecast are the multitask Gaussian Process approaches [53–56], that can be used in multi-horizon forecast applications.

3.2.3. Multi layer perceptron

In a multilayer perceptron, the structure consists of several layers of nodes interconnected through weights. The first layer is the input pattern $\mathbf{x} \in$, which is first linearly mapped into a vector \mathbf{z} through the affine transformation $\mathbf{z} = \mathbf{W}^{(1)}\mathbf{x} + \mathbf{b}^{(1)}$, where $\mathbf{W} \in \mathcal{R}^{L^{(0)} \times L^{(1)}}$ is the corresponding transformation matrix, $L^{(1)}$ is the number of nodes of the first layer and \mathbf{b} is a vector of biases. After this process, each element of vector $\mathbf{z}^{(1)}$ is transformed through a nonlinear, usually monotonically increasing function $\sigma(\cdot)$ called activation, which is typically a rectified linear function $\sigma(z) = \max(0, z)$. The process is repeated several times. The last layer has a dimension and activation that depend on the corresponding estimation task. By modeling the posterior probability distribution of the output as a function of the corresponding activation, the weights $\mathbf{W}^{(l)}, \mathbf{b}^{(lk)}$ are usually trained to maximize the training output likelihood through the well-known backpropagation algorithm [57].

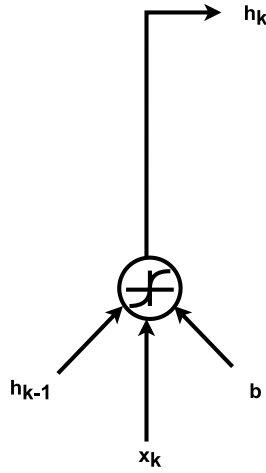


Fig. 2. RNN cell [58].

3.2.4. Recurrent neural networks

RNNs were created to handle sequential data and they are predominantly used for time series prediction, natural language processing, and speech processing [58]. They are neural networks that operate on the principle that each layer's output is saved and fed back into the network's input to forecast that layer's output. The traditional feed-forward neural network was unable to memorize the previous inputs since it allows information to flow only in the forward direction whereas an RNN contains loops, allowing information to be stored within the network. The RNN mainly consists of an input hidden layer and an output layer as demonstrated in Fig. 2. The recurrence equations of the RNN are as follows:

$$\mathbf{h}_k = \sigma(\mathbf{W}_x \mathbf{x}_k + \mathbf{W}_h \mathbf{h}_{k-1} + \mathbf{b}) \quad (6)$$

$$\mathbf{o}_k = \text{softmax}(\mathbf{W}_o \mathbf{h}_k + \mathbf{b}_o) \quad (7)$$

where \mathbf{h}_k and \mathbf{o}_k represent the vector that stores the hidden values and the output of the network at time instant k respectively. \mathbf{W}_x , \mathbf{W}_h , and \mathbf{W}_o gives the weights associated with the input, hidden, and output layer. Additionally, the biases of the recurrent and feed-forward layer are given by \mathbf{b}_r and \mathbf{b}_o . The two main disadvantages of RNN are gradient vanishing and exploding problems. When the gradient shrinks too much, the parameter changes become irrelevant, and training an RNN becomes very difficult as it does not learn the long data sequence. In the case of exploding gradients, the slope does not decay; on the contrary, it tends to grow exponentially. Large error gradients that build up during training lead to very large modifications to the weights of the neural network, which further results in longer training time and poor performance.

3.2.5. Long Short Term Memory networks

LSTM is a category of neural networks which can learn the long-term and short-term dependencies present in a sequence of data. They were developed by Hochreiter & Schmidhuber (1997) to alleviate the memory issue present in the older networks [59] due to the vanishing and exploding gradients. The LSTMs have a chain-like structure consisting of repeating units defined as cells. The LSTM has three gates namely forget gate, input gate, and output gate which facilitate the removal or addition of information to the cell state c_k as shown in Fig. 3. As the primary step the forget gate uses the output of the past hidden layer h_{k-1} and current input x_k and generates a number between 0 and 1. This gate uses a sigmoid function to decide the quantity of information that requires to be stored in the cell state. Consequently, in the next step, the input gate performs two functions. First, it uses a tanh function on h_{k-1} , and x_k gives importance to the information that is passed

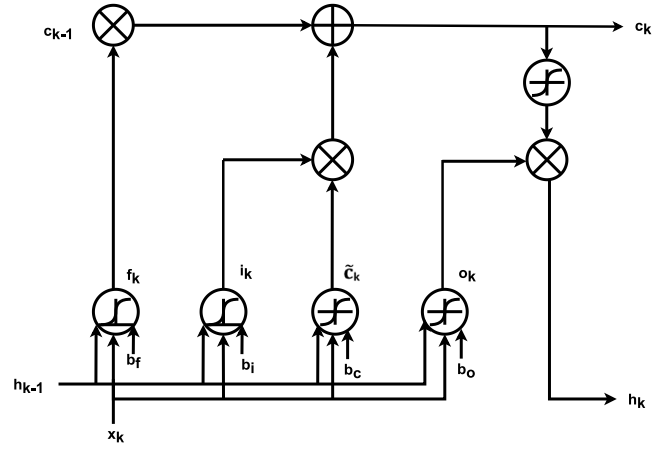


Fig. 3. LSTM cell [59].

through depending on the output values ranging between 1 and -1 . Secondly, a sigmoid function is used to decide upon the values that need to be updated. By combining the output of these steps the input gate decides whether to update the cell state c_k or not. The cell state does not suffer from the vanishing gradient phenomenon because it is passed through time without being processed by any weight matrix. The output gate on the other hand is finally used to decide which information needs to be passed forward to the next hidden state. This gate also uses a sigmoid and tanh function to determine the output. The mathematical formulation of the LSTM is given as follows:

$$\mathbf{f}_k = \sigma(\mathbf{W}_{fh} \mathbf{h}_{k-1} + \mathbf{W}_{fx} \mathbf{x}_k + \mathbf{b}_f) \quad (8)$$

$$\mathbf{i}_k = \sigma(\mathbf{W}_{ih} \mathbf{h}_{k-1} + \mathbf{W}_{ix} \mathbf{x}_k + \mathbf{b}_i) \quad (9)$$

$$\tilde{\mathbf{c}}_k = \tanh(\mathbf{W}_{ch} \mathbf{h}_{k-1} + \mathbf{W}_{cx} \mathbf{x}_k + \mathbf{b}_c) \quad (10)$$

$$\mathbf{c}_k = \mathbf{c}_{k-1} \otimes \mathbf{f}_k + \mathbf{i}_k \otimes \tilde{\mathbf{c}}_k \quad (11)$$

$$\mathbf{o}_k = \sigma(\mathbf{W}_{oh} \mathbf{h}_{k-1} + \mathbf{W}_{ox} \mathbf{x}_k + \mathbf{b}_o) \quad (12)$$

$$\mathbf{h}_k = \mathbf{o}_k \otimes \tanh(\mathbf{c}_k) \quad (13)$$

Here \mathbf{f}_k , \mathbf{i}_k , \mathbf{o}_k indicates the forget gate, input gate, and output gate. The weights of the various gates are given by \mathbf{W}_{fh} , \mathbf{W}_{fx} , \mathbf{W}_{ih} , \mathbf{W}_{ix} , \mathbf{W}_{ch} , \mathbf{W}_{cx} , \mathbf{W}_{oh} , \mathbf{W}_{ox} whereas \mathbf{b}_f , \mathbf{b}_i , \mathbf{b}_c and \mathbf{b}_o denote the biases.

3.2.6. Gated Recurrent Units

GRUs were introduced in 2014 to solve the problem of vanishing gradients that affects regular recurrent neural networks [60] but with a complexity that is lower than the one of the LSTM. The update gate and reset gate are two gates that GRUs utilize to address this issue. In general, these two vectors determine what data should be sent to the output. Information flow into memory is governed by the update gate \mathbf{i}_u . At first, the input vector \mathbf{x}_k is multiplied by the parameter weight matrices. The $k-1$ in \mathbf{h}_{k-1} denotes that it is multiplied by its weight and contains the information from the preceding unit. Then, after adding them, the results from these parameters are sent to the sigmoid activation function which outputs values between 0 and 1. If the gate vector's value is 1, the corresponding data is significant and will be used.

$$\mathbf{i}_k = \sigma(\mathbf{W}_{ih} \mathbf{h}_{k-1} + \mathbf{W}_{ix} \mathbf{x}_k + \mathbf{b}_i) \quad (14)$$

The reset gate \mathbf{f}_k , on the other hand, aids in combining the current input \mathbf{x}_k with the prior memory. After the previous hidden state has been

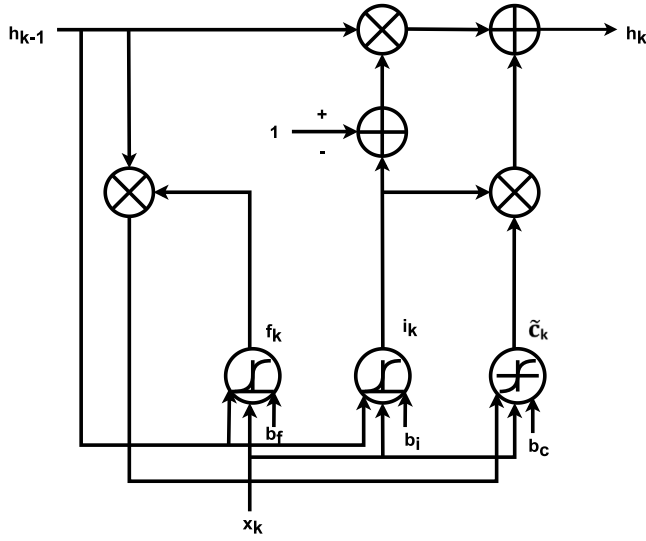


Fig. 4. GRU cell [60].

multiplied by a trainable weight, the reset vector is multiplied element-wise with it. During this process, it will be decided which information from earlier time steps should be retained with the new inputs. Simultaneously, the current input will be multiplied by a trainable weight before being added with the product of the reset vector and the prior hidden state. To achieve the vector \tilde{c}_k , the resultant vector will be passed through a non-linear activation tanh function.

$$f_k = \sigma(W_{fh}h_{k-1} + W_{fx}x_k + b_f) \quad (15)$$

$$\tilde{c}_k = \tanh(W_{cx}x_k + W_{ch}(f_k \odot h_{k-1}) + b_c) \quad (16)$$

$$h_k = (1 - i_k) \odot h_{k-1} + i_k \odot \tilde{c}_k \quad (17)$$

As a result, both of these gates are used to calculate h_k , to retain information from the past without having it fade away over time, or to discard information that is irrelevant to the forecast. The block diagram of a GRU is depicted in Fig. 4.

3.2.7. Convolutional neural networks for time series

CNNs, which were originally designed to model two-dimensional image data, may now be used to handle time series forecasting problems. A CNN can be used for prediction by applying 1D convolution to the time series that further learns the patterns from the sequence. The CNN model will train a function that maps an input consisting of a set of previous observations to an output observation. A 1D CNN model consists of a convolutional layer that works on the time series data. In certain instances, such as with very long input sequences, this is followed by a second convolutional layer. Additionally, a pooling layer is used, whose purpose is to reduce the output of the convolutional layer to the most significant feature maps. Later, following the convolutional and pooling layers is a fully connected layer that evaluates the features obtained by the model's convolutional layers.

3.3. Image based models

The image data, represented as $f(x, y)$, is constructed with one past value as the input sample, and the predictions are made for the future time horizon T . The image data in this case refers to the IR images obtained from the FLIR camera and the TSIs collected from the SRRL dataset. The CNN and SolarNet models exclusively use this information to predict future values. CNNs are powerful neural network architectures that are particularly effective for image-based tasks due

to their ability to extract hierarchical features from images. SolarNet is a CNN-based model that takes sky images as input and is specifically designed for very short-term solar irradiance forecasting. More details related to these image-based models are discussed below:

3.3.1. Convolutional neural networks for images

CNNs have been widely employed in various computer vision tasks [61] and in contrast to the MLP, they can extract spatial features like edges, texture, shape, and color patterns from the input image. These attributes are further trained and tuned to produce exceptional results in various image-processing tasks. The primary aspects that make this network robust to image translation variations are its feature learning capabilities, weight sharing, and receptive fields of the kernels. Convolutional layers are the fundamental foundation blocks of the CNN structure. These layers analyze the given image and generate the feature maps by convolving with the filters. The results from this operation are given as input to the pooling layers which aids in reducing the dimensionality of these feature maps. In general, the average or max pooling operations are used for downsampling the feature maps. All the pixels contained in the feature map are averaged during average pooling, but in max-pooling, the maximum value is calculated using the pixels in that region. Following this, the fully connected layers collect the feature maps taken from the preceding convolutional and pooling layers and make the final prediction. Further, the flattened feature maps are passed through this layer and it generates the final predicted output.

3.3.2. SolarNet

SolarNet model represented in Fig. 5 is a deep learning-based network that was developed in 2019 for very short-term solar forecasting [31]. Ground numerical measurements, satellite-based data, and sky image characteristics are the most commonly used inputs for this problem. This network used TSIs which contain images of the sky and clouds from the SRRL dataset. The TSI-800 images were chosen over other TSIs since they span a wider horizon. The original TSIs contain 352×288 pixels, but only specific regions of interest were selected to exclude areas with cloudy skies and obstacles. A 256×256 binary mask was used to crop the TSIs in a circular manner, centered at their center pixels. The TSIs were then normalized by their maximum values before being inputted into the model. The model was trained to learn latent sky image properties and estimate solar radiation automatically. In this case, only one image was utilized as input, with no endogenous parameters, to test the efficiency of automatic feature learning. The SolarNet model has 20 layers that consist of 13 layers for convolution, 5 layers that perform max-pooling layers, and 2 dense layers. The SolarNet architecture is based on the widely known extremely deep convolutional network VGG16 [62]. Here the VGG16 model is utilized as the sky image feature extractor since it is less complex than other popular networks and the performance of the SolarNet model is higher in real-time testing. The network is made up of five feature learning blocks (FLBs) and each block has either two or three convolutional layers and a single max-pooling layer. The network employs smaller kernels of size 3×3 for incorporating smaller receptive fields which give more precision. In the SolarNet-based model, the final three fully connected layers of the VGG16 were substituted with two fully connected layers and the linear activation for obtaining the predictions.

3.4. Hybrid models

Hybrid models for solar irradiance forecasting are models that combine multiple data sources and modeling techniques to improve the accuracy of predictions. These models are powerful and capitalize on the strengths of different sources of information to produce more accurate results. These models can combine past values of solar irradiance, meteorological variables, IR images, GBC images, or TSIs in various ways. Machine learning or deep learning models can also be

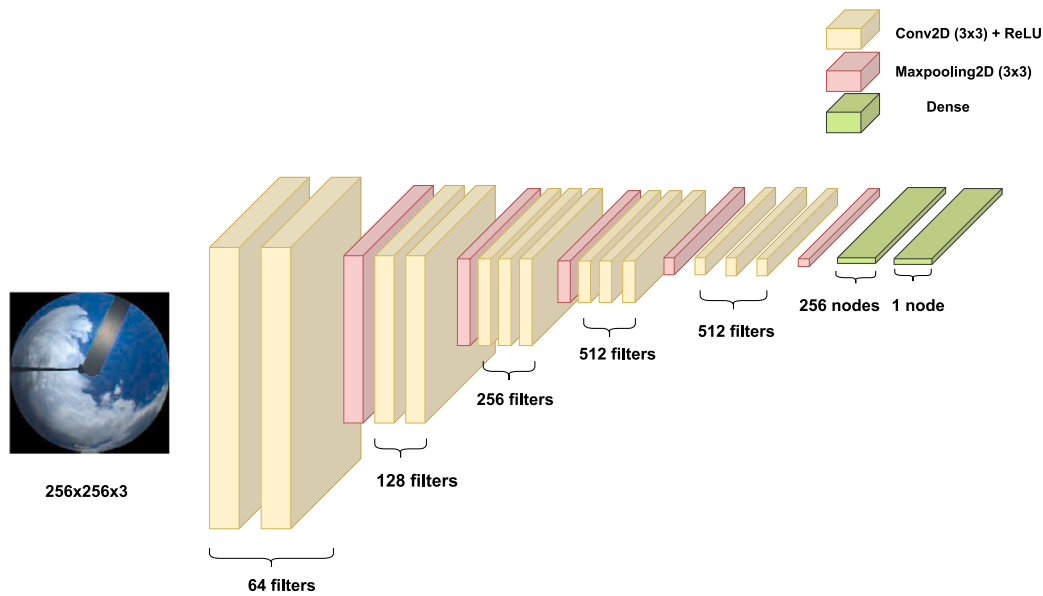


Fig. 5. SolarNet architecture [31].

combined either sequentially or in parallel to create hybrid models. By doing so, hybrid models can leverage the strengths of each individual model while minimizing their weaknesses, resulting in more accurate and reliable solar irradiance forecasts for a variety of applications. For example, the CNN-L model combines CNN and LSTM models in parallel to input past values of solar irradiance and IR images. In contrast, the SCNN-LSTM model uses Siamese CNN and LSTM models to input a sequence of TSIs and meteorological variables into the model. The upcoming sections will cover several hybrid models that have gained popularity in the literature, as well as additional models developed by authors for very short-term solar irradiance forecasting.

3.4.1. 3D-CNN

The 3D-CNN was developed as a method for automatically extracting features from GBC images for predicting solar irradiance. Since cloud motion is a major impact element in this prediction, numerous successive GBC images are used for obtaining the forecasts. It has been shown in the literature that the CNN framework with a 3D kernel (3D-CNN) can extract information from both spatial and temporal dimensions [32]. This study utilized data exclusively from the SRRL database, where the GBC images were captured by a TSI-880, and the solar irradiance values were recorded using a pyrliometer on a Sci-Tec Tracer. The GBC images had a resolution of 352×288 pixels, and a shadow band was used to protect the CCD sensor from direct sun exposure. In case of shadow band malfunction, the pointer would become inaccurate and the area surrounding the sun becomes over-exposed. Image processing techniques were employed to identify and extract the regions of interest in the GBC images, thereby eliminating the shadow band. To accomplish this, binary masks were generated with elements corresponding to the pixel locations in the image, with element values representing true or false. Only the regions identified as true, representing the region of interest were included in the input data for the 3D-CNN model. The binary masks were constructed by using a circle with a radius of 125 pixels and centering it on the image's center pixel while excluding the shadow band from the region of interest. Following the extraction of the region of interest, the image size was reduced to 251×251 pixels from the original pixel size.

After preprocessing, a 3D-CNN [39] was used for automated feature extraction of multiple sequential GBC images (Fig. 6). Here the category for classification is found by dividing the attenuation rate of direct normal irradiance (DNI) into different classes to make the automatically derived features relevant for DNI prediction. Hence this method differs

from the conventional cloud shape classification procedures [63]. The 3D-CNN model's classification criterion is based on the clear-sky index which is partitioned into 5 levels. Class 1–5 denotes cloudy, high attenuation, moderate attenuation, mild attenuation, and clear conditions, respectively [64]. During training, there was a lesser amount of GBC image samples in Classes 2 to 4. Image augmentation was implemented to address this sample imbalance. Specifically, contrast transformation was utilized as a means of augmenting the training dataset by enhancing the contrast of the newly generated images. The images of clouds at intervals $t - 2$, $t - 1$, and t were utilized as inputs, and the classification result was the class of the clear sky index at $t + i$. For the prediction, the hybrid forecasting network used the features from the fully connected layers along with the DNI values. The linear autoregressive (AR) model and the nonlinear MLP model were used as forecasting models. The parameters of the AR model were calculated using the least squares approach. The MLP used in this network consists of two hidden layers of size 30.

3.4.2. SCNN-LSTM

The SCNN-LSTM network was designed to give a 10-min ahead solar irradiance forecast [33] (Fig. 7). A Siamese CNN can extract spatial dimension characteristics from several consecutive images of the sky while retaining temporal dimension data. Then, using a concatenate layer, historical meteorological characteristics, and image features are fused after normalization. Later these combined features are sent to the LSTM network for forecasting the 10 min ahead solar irradiance. In general, a Siamese network [65] is a type of neural network composed of multiple similar subnetworks that share the same network architecture and setup, including shared weights and network parameters. The changes in the parameters are duplicated over various subnetworks throughout the training phase. The suggested Siamese convolution neural network combined the benefits of CNN and the Siamese network to select discriminative features with various levels of abstraction from sky images at various periods. The most significant distinction between this work and a standard Siamese network is that the SCNN-LSTM model does not compare the similarity of these images, hence it was not necessary to calculate the Euclidean distance between sample pairs. To train the SCNN-LSTM parameters, the prediction error (PE) was employed to assess the difference between the forecasted and actual values. This work used the DNI [66] to assess the performance of the suggested model. The data from the previous two years were collected from the NREL [67], and many tests were conducted to validate the

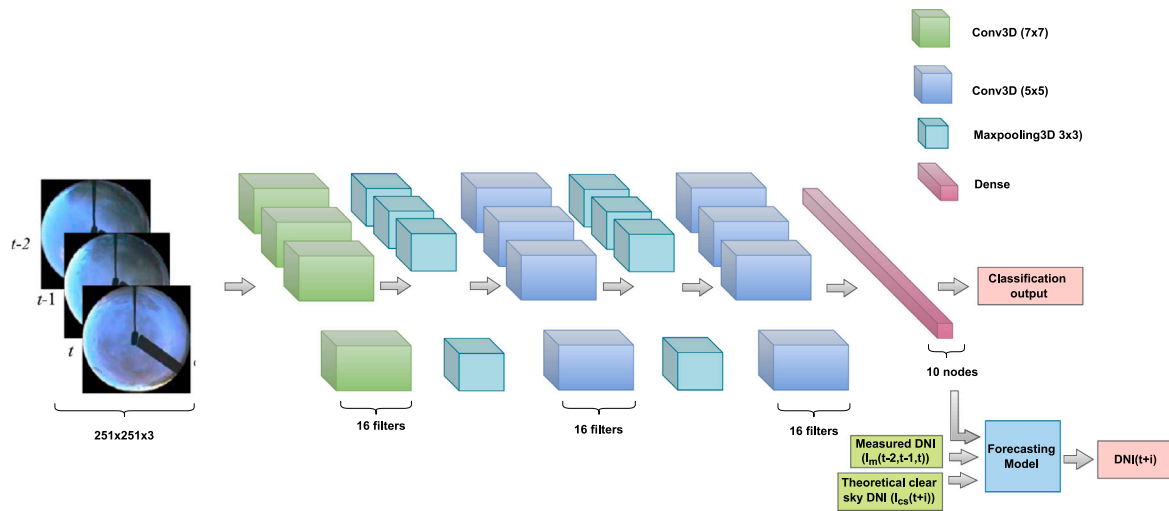


Fig. 6. 3D-CNN architecture [39].

efficacy of the approach. In the SRRL database, the meteorological variables were measured at a 1-min frequency. On the other hand, the TSI-880 captured TSIs every 10 min with a resolution of 352×288 pixels. Hence, this study utilized 10-min averages of the meteorological data as samples to maintain consistency. A fundamental clear sky model was utilized to transform DNI into a DNI clear sky index that took into consideration the effects of atmospheric, seasonal, and geographic characteristics to remove the forecast inaccuracy due to variations in the solar position [68]. The primary objective of sky image processing was to extract the region of interest from the image and remove unnecessary pixels. A binary masking technique like the one employed in 3D-CNN was used for this purpose. Additionally, a minor gradient algorithm [69] was utilized to rectify shadow-band pixels. The final input to the SCNN-LSTM consists of the region of interest of size 256×256 pixels.

3.4.3. Convolutional Long Short Term Memory Fusion Network

The proposed CNN-L model for solar radiation forecasting is a multimodal approach that integrates the spatial information from CNN, and temporal data from LSTM using deep learning [13]. The model is depicted in Fig. 8. Multimodal learning combines information from multiple sources to build an effective solution. In this approach, a parallel configuration is created to collect image features with CNN while the LSTM network extracts solar irradiance data over the period. The generated attributes are then integrated to form a generalized prediction model. The general architecture of this model is made up of three modules: (i) feature extraction block for images, (ii) feature extraction block for time-series, and (iii) multimodal feature fusion and prediction block, as shown in Fig. 8. This fusion network uses a 16-layered model that takes a single image as input and combines the 2-D image data and temporal data to estimate the predicted solar radiation. The feature extraction block for the images consists of a standard CNN with four convolutional layers, four max-pooling layers, one dropout layer, and two dense layers. The input consists of 60×80 infrared images of the cloud. The primary convolutional layer is composed of 16 3×3 filters that are succeeded by the exponential linear unit (ELU) activation function. This initial convolutional layer has a low abstraction level of features, but as it progresses through the layers, the network acquires finer details or has higher levels of abstraction. Hence, we double the number of filters compared to the previous layers; thus, the convolutional layers have 32, 64, and 128 filters. Further, we use a max-pooling layer to downsample and summarize the features in an area in the image. However, this allows the network to reduce the computational complexity and the number of parameters. The following convolutional layers have a kernel size of 3×3 , whereas

the pooling layers employ a kernel size of 2×2 . Since the smaller kernels have fewer weights, they are more computationally efficient. The output from the final pooling layer is converted to a single long continuous vector and passed to a layer with a dropout probability of 0.2 to avoid overfitting. Additionally, the obtained features are given to the fully connected/dense layers. The model has one dense layer with 64 nodes and another with 16. They were used to generate the final feature vector containing the highly discriminative features from the infrared image. The temporal block for the feature is made up of LSTM with just one layer that is given three previous time sequences of the solar irradiance data. The results of the two feature extraction modules are then linked to integrate the perceptual and temporal attributes. The feature concatenation layer combines features from several columns into a single column, and the resultant is given to two dense layers with 64 and 1 node to obtain the final prediction.

3.4.4. Multiple image convolutional long short term memory fusion network

The MICNN-L model is an extension of the CNN-L with few significant modifications, as illustrated in Fig. 9. This network takes a series of images contrary to the CNN-L model, which uses one image. Here the feature extraction module using the image has been improved to handle images that are distributed in time. The network employs similar parallel sub-networks to extract the time characteristics. Here we have a collection of images taken every 10 min, and the movement of clouds with time aids in determining whether or not they obstruct the sun. Hence the multiple frames help the prediction analysis by providing information about cloud movement and direction. Similar to the single image-based CNN-L, this model processes every image using four layers with convolutions. Later, instead of the flattened layer, the global average pooling layer is used to compute the average of each feature map from the preceding layer and to minimize dimensions. To retain the time correlations in the series of images, the feature maps from the global average pooling layer are transferred into the LSTM layers at the end. However, the temporal feature-based block remains unchanged and is equivalent to the one shown in Fig. 8. Subsequently, the output from both modules is integrated and forwarded through the dense layers. The MICNN-L structure was evaluated with a maximum of 5 image sequences. Finally, it was observed that the model that uses various consecutive images outperforms the network with a single image input.

On the other hand, MICNN-L (OF) is a replica of MICNN-L, except it uses the optical flow features as input instead of the raw images. We extract the velocity field from a sequence of consecutive images and assign a velocity to each pixel. This algorithm estimates the speed of particles in the image. It also helps to determine the direction of

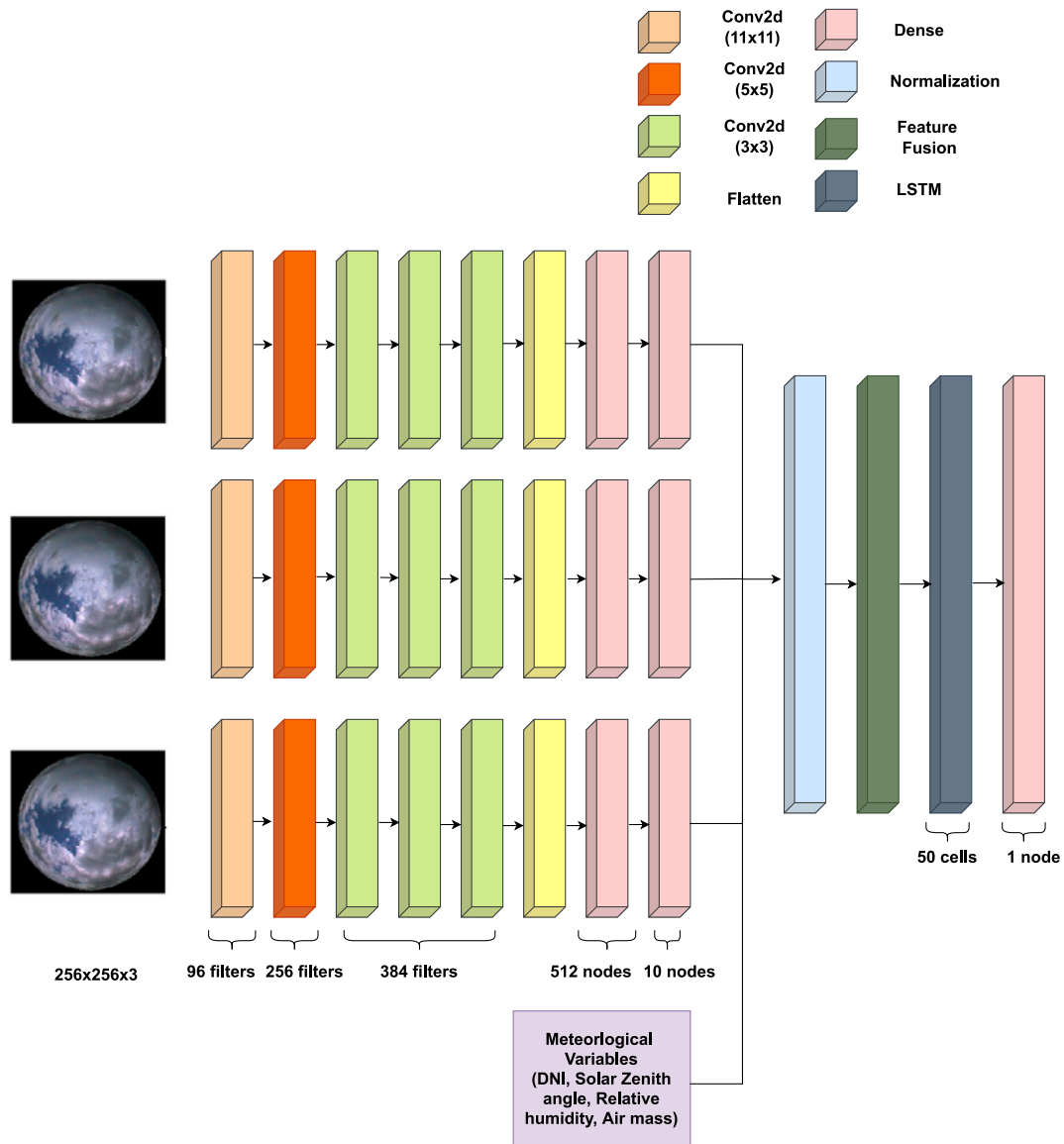


Fig. 7. SCNN-LSTM architecture [33].

motion of clouds to the sun. The motion vectors are calculated using the Horn-Schunck Optical Flow method [70]. This approach employs the motion vectors of areas changing across time to identify moving areas in an image. The calculated motion vector is a 2-dimensional vector representing the velocities and orientations of the same pixel in two successive frames. The two approximations of the algorithm are as follows: It preserves the intensity of a moving pixel across two frames, and all pixels in the current frame will be present in the succeeding frame. The MICNN-L network uses the extracted motion features for performing the predictions. This network analyzes the difference in performance while using a feature extraction in front of deep learning architecture. Though MICNN-L (OF) performs better than the rest of the approaches, it cannot outperform MICNN-L.

4. Experimental design

The proposed networks were trained and tested on cloudy days to forecast solar radiation. Previous pyranometer data is insufficient to reliably estimate solar radiation during unexpected cloud coverage or on fully overcast days. In such situations, visual information from the clouds, combined with pyranometer measurements, can be used

to reduce the forecast error. This work implemented 14 models that use different data, including time series only, image only, and a combination of both image and time series, to analyze their efficiency. Furthermore, these models were also compared with existing networks discussed in the literature, such as SolarNet, SCNN-LSTM, and 3D-CNN.

The time series models used for forecasting include SVM (Linear), SVM (Non-linear), GP (Linear), GP (Non-linear), MLP, RNN, GRU, LSTM, and CNN (R). The dataset for this experiment comprises GHI measurements obtained from the pyranometer. Forecasts were generated using the previous three readings of solar radiation. The principal preprocessing used here is data normalization, which causes the distribution to be centered around 0 and the degree of deviation from the mean as 1. The linear and non-linear SVM uses a regularization parameter C that is set to 1 along with a linear and radial basis function (RBF) kernel to produce a better fit for the data. In the case of GP, the linear model uses a linear kernel, whereas the GP (Non-linear) uses a kernel that includes the union of the linear kernel and Matérn 5/2 kernel. On the other hand, MLP, RNN, GRU, LSTM, and CNN (R) use mean square error minimization criterion along with adaptive moment optimization (Adam) [71] for training.

Next, the image-based models include the SolarNet model from the literature that used the TSI and CNN (I) model which used the

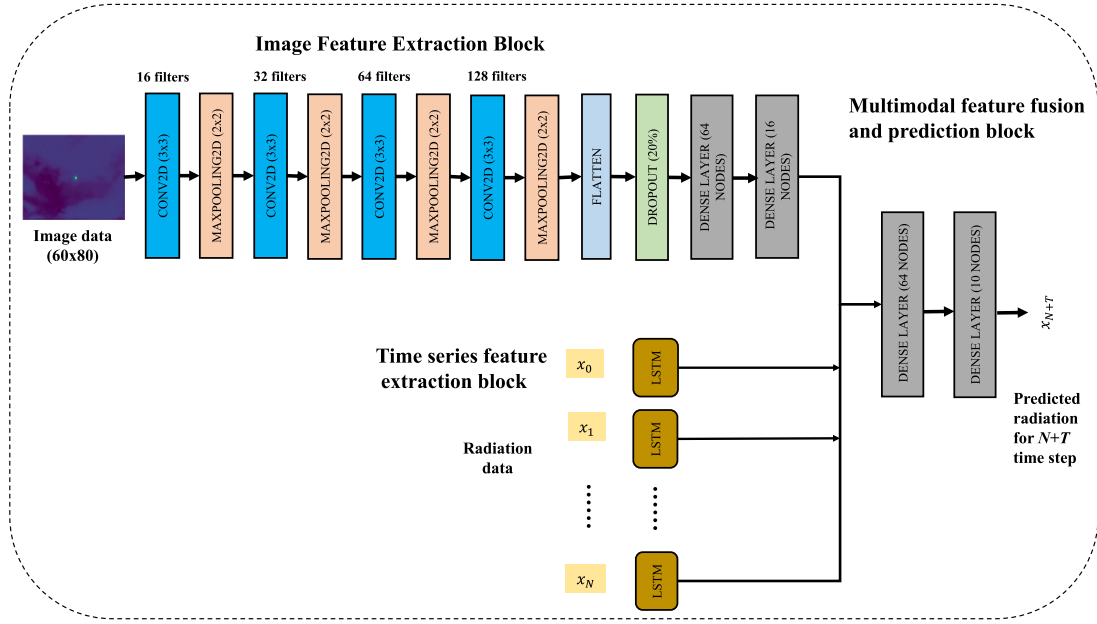


Fig. 8. CNN-L architecture [13].

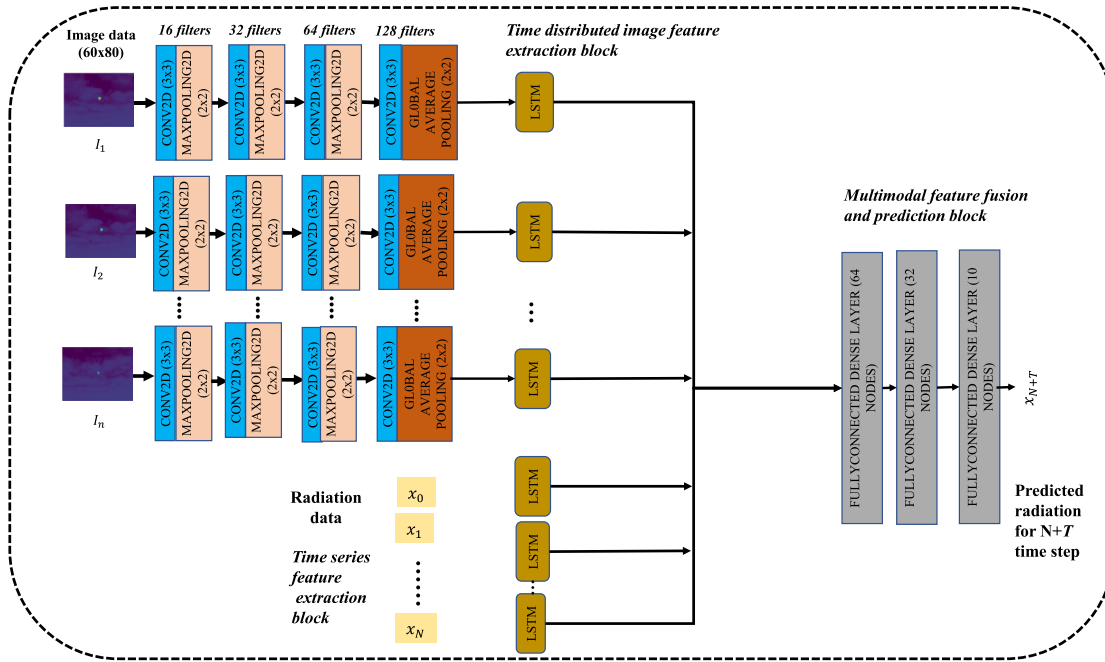


Fig. 9. MICNN-L architecture [13].

infrared images. The mini-batch stochastic gradient descent (SGD) was used to train the SolarNet model, with 14,846,273 parameters divided across 21 layers. The parameters of the model were initialized using the pre-trained weights of ImageNet. The SGD was used to minimize the objective function and update the parameters during training. The SRRL dataset, which contains both TSIs and numerical measurements, was used for case studies in this work. Ten years of numerical and image data were downloaded and processed from 2008 to 2017. The preprocessed TSI images had a resolution of 256×256 pixels. The first five years of data were used for training, the next three years for validation, and the last two years of data for testing. The entire dataset was trained for 60 epochs with a batch size of 64. The learning rate was initialized to 0.001 and was reduced by 50% when the accuracy metrics did not change more than 10 epochs. On the other hand, the

CNN (I) model used a single infrared image of the clouds for making predictions. The image was passed through a basic CNN architecture consisting of 3 layers for convolution, 3 max-pooling layers, and 2 dense layers. The convolutional layers and one of the fully connected layers used the rectified linear unit (ReLU) activation function. The structure used infra-red image data collected for 3 years (2017, 2018, and 2019) from the FLIR Lepton[®] radiometric camera. The input data had a size of 60×80 pixels and training used the minimization of the mean square error criterion.

Finally, the hybrid models were trained on both image data as well as solar radiation data. The 3D-CNN and SCNN-LSTM were models published in the literature whereas the proposed hybrid models included the CNN-L, MICNN-L, and MICNN-L (OF). For DNI prediction,

Table 2
Analysis of different models using various evaluation metrics.

Models	No. of parameters	nRMSE (%)	nMAE (%)	nMBE (%)	FS (%)	r	MAPE
Persistence	–	52.94	34.71	0.01	0	0.69	46.53
SVM (Linear)	$\mathcal{O}(N)$	43.12	26.84	5.31	18.53	0.78	40.17
SVM (Non-Linear)	$\mathcal{O}(N)$	43.87	26.33	3.80	17.11	0.77	39.22
GP (Linear)	$\mathcal{O}(N)$	43.41	29.74	10.73	18.00	0.78	54.51
GP (Non-Linear)	$\mathcal{O}(N)$	43.02	29.01	8.62	18.73	0.78	51.45
MLP	301	42.05	29.14	0.08	20.56	0.78	46.38
RNN	131	42.23	29.64	3.50	20.23	0.78	44.79
GRU	401	42.08	29.75	3.07	20.50	0.79	44.46
LSTM	491	42.06	29.89	2.62	20.54	0.79	44.78
CNN (R)	8K	42.02	28.80	0.84	20.62	0.78	44.55
CNN (I)	68K	51.88	39.86	17.79	2.00	0.72	78.01
SolarNet	15M	42.28	30.66	−4.79	17.66	0.92	–
3D-CNN	100K	38.59	25.44	3.78	24.85	0.93	–
SCNN-LSTM	5M	41.05	27.09	4.65	20.07	0.92	–
CNN-L	123K	41.70	28.58	2.32	21.23	0.79	40.92
MICNN-L (OF)	119K	37.59	26.26	7.25	28.99	0.85	35.41
MICNN-L	119K	30.12	22.85	2.47	43.10	0.94	29.20

the 3D-CNN model used features from several consecutive GBC images. The clear-sky index to be forecast is used as the label for the 3D-CNN classification model. The final forecasting model uses the fully-connected features and DNI data for DNI prediction and the model is trained throughout 32 epochs. The NREL database used in this work consists of GBC images and DNI values. The image resolution is 352×288 pixels, and the experiment uses GBC images and DNI data from January 1, 2013, to December 31, 2014. The experiments used the data from 2013 for training, 10% of data from 2014 for validation, and the rest for testing. The SCNN-LSTM also used the SRRL database with meteorological variables such as DNI, solar zenith angle, relative humidity, and air mass. The TSIs used were RGB images of resolution 352×288 pixels. The work utilized a basic clear sky model to transform the DNI into the clear sky index to reduce the forecast error resulting due to the differences in solar position. Additionally, they used image preprocessing to select the area of interest from the RGB image, and the final resolution used as the model's input was 256×256 pixels. Similarly for 3D-CNN, the source data for the experiment were gathered from 1 January 2013 to 31 December 2014. The validation set included data from January and July of 2013, whereas the testing set included data from the entire year of 2014.

The new multimodal fusion network computes the forecasts from data consisting of infrared cloud images and the pyranometric measurements from three years. The clouds moving towards the sun are highly likely to occlude the sun, making it necessary to develop a classifier based on transfer learning to distinguish between completely sunny and cloudy days. During training, the base model was created from the pre-trained MobilenetV2, initialized with the input image size of 60×80 . By applying this approach, only the last layers were retrained, leading to a faster training process. The model was subsequently trained on a small dataset by utilizing the same weights as ImageNet. Compared to other deep learning approaches, transfer learning required less training data, thereby reducing the computational burden. Various experiments were conducted to evaluate the performance of various transfer learning methods for the classifier. The models evaluated included MobilenetV2 [72], VGG16 [62], InceptionV3 [73], and ResNet50 [74]. Among these, MobileNetV2 was used to classify the fully cloudy days, and this data was fed into the proposed network as input. The developed hybrid model used the entire data from 2017 for training, 20% of data from 2018 for validation, and the rest for testing. The training was performed for 200 iterations using mini-batches of 50. The tests were divided into different sections and carried out using a laptop equipped with an NVIDIA GeForce GTX 1060 GDDR5 6.0 GB GPU, as well as high-performance computing from the University of New Mexico's Center for Advanced Research Computing.

4.1. Evaluation metrics for forecasts

In the literature, a variety of measures have been used to assess the efficacy of solar irradiance predictions. When it comes to solar irradiance forecasting, the performance is measured by evaluating the actual and forecasted solar irradiance. The following are the statistical measures that are used to evaluate the models' performance:

Normalized Root Mean Square Error (nRMSE):

$$nRMSE = \frac{\sqrt{\frac{1}{N} \sum_{i=1}^N (\hat{y}_i - y_i)^2}}{\bar{y}} \times 100\% \quad (18)$$

Normalized Mean Absolute Error (nMAE):

$$nMAE = \frac{\frac{1}{N} \sum_{i=1}^N |\hat{y}_i - y_i|}{\bar{y}} \times 100\% \quad (19)$$

Normalized Mean Bias Error (nMBE):

$$nMBE = \frac{\frac{1}{N} \sum_{i=1}^N \hat{y}_i - y_i}{\bar{y}} \times 100\% \quad (20)$$

Mean Absolute Percentage Error (MAPE):

$$MAPE = \frac{100\%}{N} \sum_{i=1}^N \left| \frac{\hat{y}_i - y_i}{\hat{y}_i} \right| \quad (21)$$

Correlation Coefficient (r):

$$r = \frac{\sum_{i=1}^N (y_i - \bar{y})(\hat{y}_i - \bar{\hat{y}})}{\sqrt{\sum_{i=1}^N (y_i - \bar{y})^2 \sum_{i=1}^N (\hat{y}_i - \bar{\hat{y}})^2}} \quad (22)$$

Forecasting Skill (FS):

$$FS = \left(1 - \frac{nRMSE_f}{nRMSE_p}\right) \times 100 \quad (23)$$

Here \hat{y}_i , $\bar{\hat{y}}$, y_i , \bar{y} , N , $nRMSE_f$ and $nRMSE_p$ represent the network result, an average of network result, actual result, an average of the actual result, number of samples, nRMSE of the forecast model and nRMSE of persistence model respectively.

5. Results and discussion

We first analyze the results and choice of the methodology used to discriminate cloudy from sunny days. In this application, MobileNetV2 shows the best performance of all alternative transfer learning models such as VGG16, InceptionV3, and ResNet50. This model was able to categorize the cloudy images with an accuracy of 99.23% after being trained on a limited dataset employing similar weights from the ImageNet database. Later, the developed hybrid model was trained and tested on overcast days. The training objective involved decreasing the

mean square error by utilizing the Adam optimizer. This optimizer achieved the best results compared to several different optimizers, including SGD, RMSprop [75], and AdaDelta [76]. It adapts the learning rate for each weight of the network by estimating the first and second moments of the gradient. Consequently, Adam uses the principles of SGD and RMSprop and outperforms other optimizers.

Table 2 illustrates the 10-min ahead forecasting results of the different networks using various evaluation measures. Here six distinct statistical metrics often employed in the literature were discussed to analyze the different networks' effectiveness. We calculate the MAPE, r , nRMSE, nMAE, nMBE, and FS for the proposed and other existing models during overcast days. Here the persistence model is a simple prediction model frequently used to assess the performance of various prediction models. This study assumes that the predicted value in the future time is the same as the current value. Forecasting skill is a metric that compares a chosen model to a reference model, i.e., the persistence model. The table shows that the presented MICNN-L and MICNN-L (OF) structures surpass all the compared models with the highest FS of 43.10 and 28.99. The section containing time series models MLP, RNN, GRU, LSTM, and CNN (R) obtained similar FS ranging from 20.23 to 20.62. The image-based approaches, such as CNN (I) and SolarNet, showed one of the lowest FS values. This denotes that robust solar radiation predictions can only be made with one RGB or infrared image. The hybrid models like SolarNet, 3D-CNN, and SCNN-LSTM had an FS that was slightly higher than the time series models, and amongst them, 3D-CNN achieved the highest FS of 24.85. Hence the use of multimodal systems for forecasting gives a better performance than unimodal networks. Further, in terms of nRMSE values, most models show an error between 41.05 and 52.94. The MICNN-L, MICNN-L (OF), and 3D-CNN models achieved the least nRMSE values, 30.12, 37.59, and 38.59, respectively. It shows that visual information from consecutive images helps minimize the forecast error.

The presented table facilitates an assessment of the computational complexity and capacity of the distinct models by representing the number of trainable parameters attributed to each method. Models with a larger number of trainable parameters generally have a greater capacity to learn complex patterns in data, leading to improved accuracy and performance. Furthermore, such models can better adapt to diverse and nuanced representations of input data. However, they may be more prone to overfitting, and their training and evaluation require more computational resources. The added complexity in training also makes it more challenging to identify the optimal set of parameter values that can accurately capture the underlying patterns in the data. The number of trainable parameters in SVMs and GPs varies depending on the implementation and kernel function utilized. Specifically, in SVMs, the number of trainable parameters is dependent on both the number of features in the input data and the chosen kernel function. Meanwhile, in GPs, the number of trainable parameters relies on the selection of kernel function as well as the number of hyperparameters used to define the kernel. In this study, both SVMs and GPs exhibit computational complexity of order $\mathcal{O}(N)$, where N represents the number of samples. MLPs and RNNs, which possess only a few trainable parameters, are generally less complex than advanced recurrent and convolutional networks. Conversely, LSTMs, with 491 trainable parameters, are more complex due to their intricate gating mechanisms and were designed to process sequential data over extended periods of time. Finally, GRUs, which possess 401 trainable parameters, are like LSTMs but feature fewer parameters and can prove effective for tasks requiring sequential processing and short-term memory.

The number of trainable parameters in a CNN depends on factors such as the number and size of the convolutional and fully connected layers used. For instance, the 1D-CNN that processes time series data has approximately eight times fewer parameters than a 2D-CNN that processes infrared images. However, models with a higher number of parameters, such as the SolarNet model that solely uses images and

possesses 15 million parameters, are likely to suffer from overfitting and exhibit degraded performance. In contrast, hybrid models like 3D-CNN, MICNN-L (OF), and MICNN-L require only 100-119K parameters, enabling them to learn and generalize complex data patterns without sacrificing performance. Therefore, achieving a balance between the number of parameters and considering the trade-offs between model complexity, performance, and computational efficiency is essential.

Fig. 10 presents a scatter plot of the predicted and measured solar irradiance values for the different models. The correlation coefficient r for the 14 developed and tested models ranged from 0.6952 to 0.9416. The highest value corresponded to the proposed MICNN-L with $r = 0.9416$. Moreover, the scatterplot also shows that the degree of dispersion of MICNN-L is lesser than in other models. For comparison, the authors of the SolarNet, 3D-CNN, and SCNN-LSTM reported values of 0.9226, 0.9356, and 0.9274 in their respective papers.

Since the data used in all the experiments consists only of cloudy days, the measured performances show a very high uncertainty. Indeed, the nMAE among all models ranges between 22% and 40%, and the nRMSE is between 53% and 30%. This is consistent with the previous literature on solar forecasts. It is remarkable that if only deep learning techniques are used, the maximum drops to 40% and the maximum MAE drops to 42% if we exclude the CNN (I), which did not show good results. The MAPE among deep learning methods falls between 30 and 46%. Thus, this paper shows that the use of images and deep learning reduces forecast uncertainty.

The uncertainty increases with the time horizon. For example, in previous experiments [13] we show that this uncertainty, when using fusion methods, ranges from 5% to 8%, and between 5% and 13% when the DL methods use images only. Thus, in order to reduce the impact of the forecast uncertainty it is advisable to use a multi-horizon setup, in order to correct imbalances between power demand and generation as often as possible. There are several ways to measure the impact of forecast uncertainty in renewable energy management. Impact measurement and assessment are fundamental for grid operators to properly and safely balance power, for unbalances may result in a significant economic loss (if a significant quantity of regulation reserves are needed to reduce the unbalance to zero) and even major grid failures (if an imbalance event occurs). This assessment is beyond the scope of this paper and our technical capabilities. Nevertheless, there are several works that measure the limitations derived from these uncertainties. For example, in [77], this analysis is done in detail (see also [78]), plus the authors introduce mitigation strategies. Violations of the control performance standard (CPS) established by the North American Electric Reliability Corporation (NERC) are used as impact metrics. In particular, the CPS2 is violated when the area control error is over a certain threshold. The study shows that, for a study in the Arizona Public Service System, an imperfect forecasting structure with an average MAE of 2% can save up to \$3M per year, where short-term forecast uncertainty has little impact on operation costs. A more recent work [79] assesses directly the economic impact of the uncertainty in the short-term forecast in terms of economic savings with respect to the perfect forecast with certain particular deep learning structures. While a perfect forecast can produce a saving of about 15%, the use of an LSTM or a simple MLP can save about 12%. These studies show that data-based forecast improves the efficiency of renewable energies over the persistence-based method and, therefore, the research and application of machine learning in the solar forecast is justified and will produce significant improvements in the future.

6. Conclusion

This work presents a comprehensive analysis of short-term solar irradiance forecasting models based on the input data type. Here the forecasting methods are classified as time series-based models, image-based models, and hybrid models. A detailed description, implementation, and performance analysis of several state-of-the-art models, including

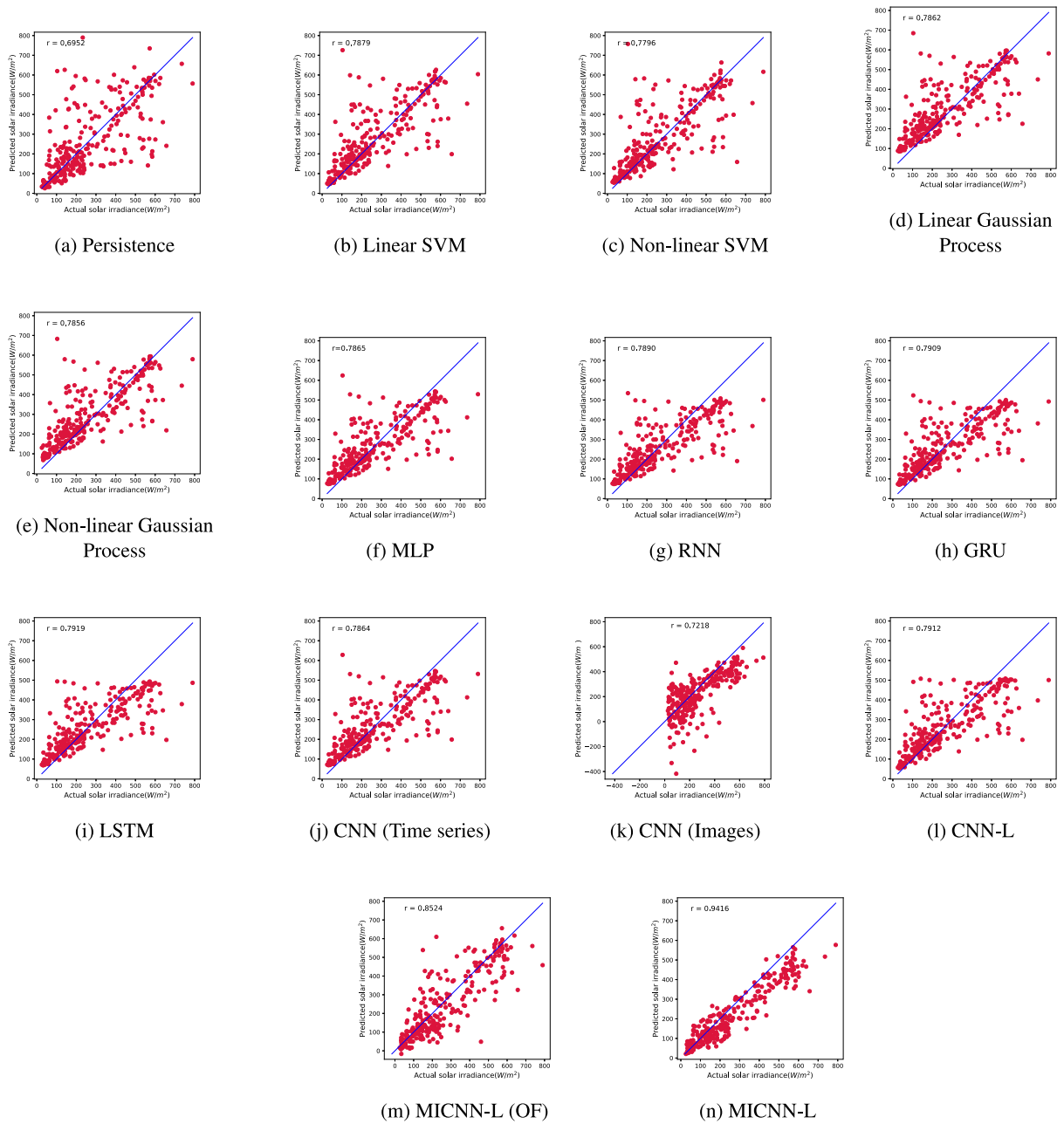


Fig. 10. Scatter plot of the measured and predicted solar irradiance using different approaches.

SVM, GP, MLP, GRU, RNN, LSTM, and CNN, are provided in this study. Moreover, three hybrid models, CNN-L, MICNN-L, and MICNN-L (OF), are also proposed and evaluated against existing approaches. Further comparisons are made against many deep learning models in the literature, such as SolarNet, 3D-CNN, and SCNN-LSTM. The deep architecture of these models makes it easier to extract high-level, non-linear information from the solar data.

The time series-based models discussed in this paper are SVM, GP, MLP, RNN, LSTM, GRU, and CNN. Here SVM and GP show similar performance, and their non-linear version shows slight improvement compared to the linear version. This indicates the non-linear nature of the data and the requirement of non-linear kernels to map the data into a high dimensional space for better prediction. The recurrent models, such as the LSTMs, can handle long-term dependencies in the time series data. The GRUs, on the other hand, is computationally more efficient since it uses less memory and is much faster. Both were designed to address the vanishing gradient problem present in the RNN. In terms

of solar irradiance data used in this study, RNN, GRU, and LSTM give a comparable performance when assessing their nRMSE, nMAE, FS, and MAPE. The deep networks such as MLP and CNN (R) that use solar radiation time series do not show further improvement. Hence, this shows that time series information is insufficient in predicting solar irradiance accurately. Next, in the case of image-based models, 2D-CNN works more efficiently since its design exploits the spatial relationships in data, and here it works well on the TSI images (SolarNet) compared to the IR images (CNN (I)).

Finally, all the hybrid methods can extract spatial and temporal data and use a series, parallel or cascaded structure per their design. They are computationally more expensive than the other methods but have high efficiency. The 3D-CNN model uses a cascaded structure in which the visual information is extracted from a sequence of GBC images using the 3D-CNN. It was further combined with the measured DNI and theoretical clear sky DNI and passed into the MLP model for making

the predictions. The main disadvantage of this model is that it fails to capture the temporal relationship between the sequence of images.

SCNN-LSTM also uses a cascaded model that extracts the visual information using the 2D-CNN-based siamese networks. It helps to find the similarity between the sequence of images by comparing their feature vectors. The output of this network is combined with the meteorological variables and passed through an LSTM for forecasting. Here, the meteorological variables do not undergo any feature extraction before concatenating with the visual features. Thus the temporal information present in these variables is not extracted adequately, which results in a decrease in performance compared to the other hybrid models.

Further, three hybrid models such as CNN-L, MICNN-L, and MICNN-L (OF), are also proposed in this study. The CNN-L models utilize a parallel structure and combine the information from a single infra-red image and past solar irradiance values to make the predictions. It gives competitive performance compared to the other models present in the literature. The main disadvantage of this model is that it uses only a single infra-red image for visual information extraction. The network has competitive nRMSE, nMAE, and nMBE values compared to other hybrid models, but it has a low r value since it cannot capture the information related to the motion of the clouds. The best-performing models were MICNN-L and MICNN-L (OF), which had a forecasting skill of 43.10 and 28.99, respectively. Both models used a sequence of infra-red images and past values of solar irradiance. The MICNN-L(OF) used the optical flow-based feature extraction before passing the image data into the deep learning model. Nevertheless, compared to the MICNN-L model, it did not show any improvement; on the contrary, MICNN-L (OF) showed a significant decrease in performance compared to MICNN-L. The external feature extraction decreased the performance of the hybrid model. Hence this concludes that no external feature extraction is required for the deep learning hybrid model for efficient solar irradiance forecasting.

Additionally, a forecasting model's efficacy fluctuates with the weather. The proposed models have been tested in cloudy conditions and show better performance than the other approaches. Conclusively, deep learning techniques have great potential in solving complex time-series forecasting problems, and deep hybrid models enhance performance compared to unimodal approaches.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request

Acknowledgments

This research was rendered by NSF EPSCoR, United States award OIA-1757207 and King Felipe VI endowed Chair. The authors would like to thank the UNM Center for Advanced Research Computing, for the high-performance computing resources utilized in this study.

References

- [1] Sampath Kumar D, Yaghi G, Kashyap M, Srinivasan D. Solar irradiance resource and forecasting: A comprehensive review. *IET Renew Power Gener* 2020;14:1641–56, (15).
- [2] Anvari M, Lohmann G, Wächter M, Milan P, Lorenz E, Heinemann D, Tabar MRR, Peinke J. Short term fluctuations of wind and solar power systems. *New J Phys* 2016;18(6):063027.
- [3] Zang H, Liu L, Sun L, Cheng L, Wei Z, Sun G. Short-term global horizontal irradiance forecasting based on a hybrid CNN-LSTM model with spatiotemporal correlations. *Renew Energy* 2020;160:26–41.
- [4] Kumari P, Toshniwal D. Deep learning models for solar irradiance forecasting: A comprehensive review. *J Clean Prod* 2021;318:128566.
- [5] Wan C, Zhao J, Song Y, Xu Z, Lin J, Hu Z. Photovoltaic and solar power forecasting for smart grid energy management. *CSEE J Power Energy Syst* 2015;1(4):38–46.
- [6] Verbois H, Huva R, Rusydi A, Walsh W. Solar irradiance forecasting in the tropics using numerical weather prediction and statistical learning. *Sol Energy* 2018;162:265–77.
- [7] Diagne M, David M, Lauret P, Boland J, Schmutz N. Review of solar irradiance forecasting methods and a proposition for small-scale insular grids. *Renew Sustain Energy Rev* 2013;27:65–76.
- [8] Yang D. Comment: Operational aspects of solar forecasting. *Sol Energy* 2020;210:38–40.
- [9] Atique S, Noureen S, Roy V, Bayne S, Macfie J. Time series forecasting of total daily solar energy generation: A comparative analysis between ARIMA and machine learning techniques. In: 2020 IEEE green technologies conference (GreenTech). 2020, p. 175–80.
- [10] Makridakis S, Spiliotis E, Assimakopoulos V. Statistical and Machine Learning forecasting methods: Concerns and ways forward. *PLoS One* 2018;13(3):e0194889.
- [11] Mammoli A, Ellis A, Menicucci A, Willard S, Caudell T, Simmins J. Low-cost solar micro-forecasts for PV smoothing. In: 2013 1st IEEE conference on technologies for sustainability (SusTech). IEEE; 2013, p. 238–43.
- [12] Terrén-Serrano G, Martínez-Ramón M. Kernel learning for intra-hour solar forecasting with infrared sky images and cloud dynamic feature extraction. *Renew Sustain Energy Rev* 2023;175:113125.
- [13] Ajith M, Martínez-Ramón M. Deep learning based solar radiation micro forecast by fusion of infrared cloud images and radiation data. *Appl Energy* 2021;294:117014.
- [14] Lin J, Li H, et al. A short-term PV power forecasting method using a hybrid Kmeans-GRA-SVR model under ideal weather condition. *J Comput Commun* 2020;8(11):102.
- [15] Yadav AP, Behera L. Solar Radiation forecasting using neural networks and Wavelet Transform. *IFAC Proc Vol* 2014;47(1):890–6.
- [16] Lubbe F, Maritz J, Harms T. Evaluating the potential of Gaussian process regression for solar radiation forecasting: A case study. *Energies* 2020;13(20):5509.
- [17] Leva S, Dolara A, Grimaccia F, Mussetta M, Ogliari E. Analysis and validation of 24 hours ahead neural network forecasting of photovoltaic output power. *Math Comput Simulation* 2017;131:88–100, 11th International Conference on Modeling and Simulation of Electric Machines, Converters and Systems.
- [18] Rahman A, Srikumar V, Smith AD. Predicting electricity consumption for commercial and residential buildings using deep recurrent neural networks. *Appl Energy* 2018;212:372–85.
- [19] Li Q, Wu Z, Ling R, Tan M. Echo state network-based spatio-temporal model for solar irradiance estimation. *Energy Procedia* 2019;158:3808–13.
- [20] Bengio Y, Simard P, Frasconi P. Learning long-term dependencies with gradient descent is difficult. *IEEE Trans Neural Netw* 1994;5(2):157–66.
- [21] Pascanu R, Mikolov T, Bengio Y. On the difficulty of training recurrent neural networks. In: International conference on machine learning. PMLR; 2013, p. 1310–8.
- [22] Ribeiro AH, Tiels K, Aguirre LA, Schön T. Beyond exploding and vanishing gradients: analysing RNN training using attractors and smoothness. In: International conference on artificial intelligence and statistics. PMLR; 2020, p. 2370–80.
- [23] Ge Y, Nan Y, Bai L. A hybrid prediction model for solar radiation based on long short-term memory, empirical mode decomposition, and solar profiles for energy harvesting wireless sensor networks. *Energies* 2019;12(24):4762.
- [24] Yu Y, Cao J, Zhu J. An LSTM short-term solar irradiance forecasting under complicated weather conditions. *IEEE Access* 2019;7:145651–66.
- [25] Qing X, Niu Y. Hourly day-ahead solar irradiance prediction using weather forecasts by LSTM. *Energy* 2018;148:461–8.
- [26] Yan K, Shen H, Wang L, Zhou H, Xu M, Mo Y. Short-term solar irradiance forecasting based on a hybrid deep learning methodology. *Information* 2020;11(1):32.
- [27] Sun Y, Szűcs G, Brandt AR. Solar PV output prediction from video streams using convolutional neural networks. *Energy Environ Sci* 2018;11(7):1811–8.
- [28] Wang K, Li K, Zhou L, Hu Y, Cheng Z, Liu J, Chen C. Multiple convolutional neural networks for multivariate time series prediction. *Neurocomputing* 2019;360:107–19.
- [29] Yeom J-M, Park S, Chae T, Kim J-Y, Lee CS. Spatial assessment of solar radiation by machine learning and deep neural network models using data provided by the COMS MI geostationary satellite: A case study in South Korea. *Sensors* 2019;19(9):2082.
- [30] Moncada A, Richardson Jr W, Vega-Avila R. Deep learning to forecast solar irradiance using a six-month UTSA SkyImager dataset. *Energies* 2018;11(8):1988.
- [31] Feng C, Zhang J. SolarNet: A sky image-based deep convolutional neural network for intra-hour solar forecasting. *Sol Energy* 2020;204:71–8.
- [32] Ji S, Xu W, Yang M, Yu K. 3D convolutional neural networks for human action recognition. *IEEE Trans Pattern Anal Mach Intell* 2012;35(1):221–31.
- [33] Zhu T, Guo Y, Li Z, Wang C. Solar radiation prediction based on convolution neural network and long short-term memory. *Energies* 2021;14(24):8498.

- [34] Shan S, Li C, Ding Z, Wang Y, Zhang K, Wei H. Ensemble learning based multi-modal intra-hour irradiance forecasting. *Energy Convers Manage* 2022;270:116206.
- [35] Terrén-Serrano G, Bashir A, Estrada T, Martínez-Ramón M. Girasol, a sky imaging and global solar irradiance dataset. *Data Brief* 2021;35:106914.
- [36] Feng C, Yang D, Hodge B-M, Zhang J. OpenSolar: Promoting the openness and accessibility of diverse public solar datasets. *Sol Energy* 2019;188:1369–79.
- [37] Zhang J, Verschae R, Nobuhara S, Lalonde J-F. Deep photovoltaic nowcasting. *Sol Energy* 2018;176:267–76.
- [38] Limouni T, Yaagoubi R, Bouziane K, Guissi K, Baali EH. Accurate one step and multistep forecasting of very short-term PV power using LSTM-TCN model. *Renew Energy* 2023;205:1010–24.
- [39] Zhao X, Wei H, Wang H, Zhu T, Zhang K. 3D-CNN-based feature extraction of ground-based cloud images for direct normal irradiance prediction. *Sol Energy* 2019;181:510–8.
- [40] Boser BE, Guyon IM, Vapnik VN. A training algorithm for optimal margin classifiers. In: *Proceedings of the fifth annual workshop on computational learning theory*. 1992, p. 144–52.
- [41] Guyon I, Boser B, Vapnik V. Automatic capacity tuning of very large VC-dimension classifiers. *Adv Neural Inf Process Syst* 1992;5.
- [42] Schölkopf P, Burges C, Vapnik V. Extracting support data for a given task. In: *Proceedings of the 1st international conference on knowledge discovery & data mining*. 1995, p. 252–7.
- [43] Cortes C, Vapnik V. Support-vector networks. *Mach Learn* 1995;20(3):273–97.
- [44] Schölkopf B, Burges C, Vapnik V. Incorporating invariances in support vector learning machines. In: *International conference on artificial neural networks*. Springer; 1996, p. 47–52.
- [45] Vapnik V. *The nature of statistical learning theory*. Springer science & business media; 1999.
- [46] Shawe-Taylor J, Cristianini N, et al. *Kernel methods for pattern analysis*. Cambridge University Press; 2004.
- [47] Aizerman MA. Theoretical foundations of the potential function method in pattern recognition learning. *Autom Remote Control* 1964;25:821–37.
- [48] Schölkopf B, Herbrich R, Smola AJ. A generalized representer theorem. In: *International conference on computational learning theory*. Springer; 2001, p. 416–26.
- [49] Vapnik V, Golowich S, Smola A. Support vector method for function approximation, regression estimation and signal processing. *Adv Neural Inf Process Syst* 1996;9.
- [50] Xu S, An X, Qiao X, Zhu L, Li L. Multi-output least-squares support vector regression machines. *Pattern Recognit Lett* 2013;34(9):1078–84.
- [51] Ruiz C, Alaiz CM, Dorronsoro JR. Multitask support vector regression for solar and wind energy prediction. *Energies* 2020;13(23):6308.
- [52] Rasmussen CE. Gaussian processes in machine learning. In: *Summer school on machine learning*. Springer; 2003, p. 63–71.
- [53] Williams C, Klanke S, Vijayakumar S, Chai KM. Multi-task Gaussian process learning of robot inverse dynamics. In: *Advances in neural information processing systems*. 2009, p. 265–72.
- [54] Bonilla EV, Chai KM, Williams C. Multi-task Gaussian process prediction. In: *Platt JC, Koller D, Singer Y, Roweis ST, editors. Advances in neural information processing systems 20*. Curran Associates, Inc.; 2008, p. 153–60.
- [55] Álvarez MA, Lawrence ND. Computationally efficient convolved multiple output Gaussian processes. *J Mach Learn Res* 2011;12(May):1459–500.
- [56] García-Hinde Ó, Martínez-Ramón M, Gómez-Verdejo V. A conditional one-output likelihood formulation for multitask Gaussian processes. *Neurocomputing* 2022;257–70.
- [57] Werbos PJ. *Beyond regression: New tools for prediction and analysis in the behavioral sciences* (Ph.D. thesis), Harvard University; 1974.
- [58] Rumelhart DE, Hinton GE, Williams RJ. Learning representations by back-propagating errors. *Nature* 1986;323(6088):533–6.
- [59] Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Comput* 1997;9:1735–80.
- [60] Cho K, van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics; 2014, p. 1724–34.
- [61] Lecun Y, Bengio Y, Hinton G. Deep learning. *Nature Cell Biol* 2015;521(7553):436–44.
- [62] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. 2014, arXiv preprint arXiv:1409.1556.
- [63] Zhu T, Wei H, Zhao X, Zhang K, Fang S. A method of cloud classification based on DNI. In: *2016 35th Chinese control conference (CCC)*. IEEE; 2016, p. 4155–60.
- [64] Martínez-Chico M, Batlles F, Bosch J. Cloud classification in a Mediterranean location using radiation data and sky images. *Energy* 2011;36(7):4055–62.
- [65] Ni C, Wang D, Vinson R, Holmes M, Tao Y. Automatic inspection machine for maize kernels based on deep convolutional neural networks. *Biosyst Eng* 2019;178:131–44.
- [66] Law EW, Prasad AA, Kay M, Taylor RA. Direct normal irradiance forecasting and its application to concentrated solar thermal output forecasting—A review. *Sol Energy* 2014;108:287–307.
- [67] Achleitner S, Kamthe A, Liu T, Cerpa AE. SIPS: Solar irradiance prediction system. In: *IPSN-14 Proceedings of the 13th international symposium on information processing in sensor networks*. IEEE; 2014, p. 225–36.
- [68] Zhu T, Wei H, Zhao X, Zhang C, Zhang K. Clear-sky model for wavelet forecast of direct normal irradiance. *Renew Energy* 2017;104:1–8.
- [69] Zhu X, Zhou H, Zhu T, Jin S, Wei H. Pre-processing of ground-based cloud images in photovoltaic system. *Autom Electr Power Syst* 2018;42:140–5.
- [70] Horn BK, Schunck BG. Determining optical flow. *Artificial Intelligence* 1981;17(1–3):185–203.
- [71] Kingma DP, Ba J. Adam: A method for stochastic optimization. 2014, arXiv preprint arXiv:1412.6980.
- [72] Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M, Adam H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. 2017, arXiv preprint arXiv:1704.04861.
- [73] Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the inception architecture for computer vision. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, p. 2818–26.
- [74] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, p. 770–8.
- [75] Dauphin Y, De Vries H, Bengio Y. Equilibrated adaptive learning rates for non-convex optimization. *Adv Neural Inf Process Syst* 2015;28.
- [76] Zeiler MD. Adadelta: an adaptive learning rate method. 2012, arXiv preprint arXiv:1212.5701.
- [77] Ela E, Diakov V, Ibanez E, Heaney M. Impacts of variability and uncertainty in solar photovoltaic generation at multiple timescales. Technical report, Golden, CO (United States): National Renewable Energy Lab.(NREL); 2013.
- [78] Mills A, Ahlstrom M, Brower M, Ellis A, George R, Hoff T, Kroposki B, Lenox C, Miller N, Stein J, et al. Understanding variability and uncertainty of photovoltaics for integration with the electric power system (LBNL-2855E). Berkeley, CA: Lawrence Berkeley National Laboratory; 2009.
- [79] Husein M, Chung I-Y. Impact of solar power and load demand forecast uncertainty on the optimal operation of microgrid. In: *2019 IEEE PES/IAS PowerAfrica*. 2019, p. 199–203. <http://dx.doi.org/10.1109/PowerAfrica.2019.8928924>.