

Convolutional neural networks for intra-hour solar forecasting based on sky image sequences

Cong Feng^{a,b,*}, Jie Zhang^{b,c}, Wenqi Zhang^d, Bri-Mathias Hodge^{d,e,f}

^a Power Systems Engineering Center, National Renewable Energy Laboratory, United States

^b Department of Mechanical Engineering, The University of Texas at Dallas, United States

^c Department of Electrical and Computer Engineering, The University of Texas at Dallas, United States

^d Grid Planning and Analysis Center, National Renewable Energy Laboratory, United States

^e Department of Electrical, Computer & Energy Engineering, University of Colorado Boulder, United States

^f Renewable and Sustainable Energy Institute, University of Colorado Boulder, United States

ARTICLE INFO

Keywords:

Deep learning
CNN
Solar forecasting
Sky image sequence
Computer vision

ABSTRACT

Accurate and timely solar forecasts play an increasingly critical role in power systems. Compared to longer forecasting timescales, very short-term solar forecasting has lagged behind in both research and practice. In this paper, we propose deep convolutional neural networks (CNNs) to provide operational intra-hour (10-minute-ahead to 60-minute-ahead) solar forecasts. We develop two CNN structures inspired by a widely-used CNN architecture. The CNNs are tailored to our solar forecasting regression tasks and rely solely on sky image sequences. Case studies based on six years of data (over 150,000 data points) demonstrate that the best CNN model has forecast skill scores of 20%–39% over the naive persistence of cloudiness benchmark, even at these very short timescales. The CNNs also have consistently superior performance when compared to shallow machine learning models with meteorological predictors, where the improvement averages around 7%. The sensitivity analyses show that the sky image length, resolution, and weather conditions have impacts on the deep learning model accuracy. In our intra-hour problem with specific setups, two sky images with a 10-minute 128×128 resolution yield the most accurate forecasts. Current limitations, future work, and deployment challenges and solutions are also discussed.

1. Introduction

The importance of accurate solar forecasts has been recognized, which has led to a considerable number of solar forecasting publications. According to the Google Scholar database, there are over 22,000 records on solar irradiance/power forecasting published in 2020. Existing solar forecasting methods can be categorized in various ways. For example, in one of the most popular review papers, Inman et al. [1] classified solar forecasting techniques into five groups, including regressive methods, artificial intelligence methods, remote sensing models, numerical weather prediction (NWP), and local sensing. Similar categorization was proposed by Yang et al. [2] and Diagne et al. [3]. Depending on the use of analytical equations, solar forecasting models were divided into “white box”, “black box”, or “gray box” models [4]. White box models rely on physical laws to model the photovoltaic (PV) system. Black box methods refer to statistical and machine learning methods that capture numerical patterns within the data. A gray box

model is a combination of both methods. It is observed that algorithm principles, input sources, and forecasting timescales are the most widely accepted classification criteria for solar forecasting techniques.

The development of machine learning and sensing technologies has blurred the boundaries between different categories of models. For instance, NWP outputs and satellite-based features are always included as inputs to regressive and machine learning models for intra-day and day-ahead solar forecasting. On the other hand, regressive and machine learning methods are also combined with local sensing networks for better intra-day solar forecasting. NWP outputs and in-situ meteorological measurements are the most widely-used inputs to the machine learning models. For example, NWP and satellite data were used as exogenous input to forecast 1-hour-ahead (1HA) to 6HA global horizontal irradiance (GHI) [5]. Similarly, satellite-based cloud motion features were integrated with NWP by Wolff et al. [6] to improve the forecasting accuracy. Four parameters, including the total cloud cover, the ambient temperature, the weather condition, and the plane

* Corresponding author at: Power Systems Engineering Center, National Renewable Energy Laboratory, United States.
E-mail address: cong.feng@nrel.gov (C. Feng).

of array irradiance clear sky model, along with the PV AC power rating, were used for 12HA solar power forecasting [7]. In addition, measurements of the neighboring PV systems were also informative as they could provide correlations for 5-minute-ahead (5MA) to 8MA solar forecasting models, which showed a 6% forecast skill score over persistence of cloudiness [8].

Sky images have been proved to be another informative source of input. However, sky image-based solar forecasting seemingly lags behind other methods, both in practice and in research. One reason could be that sky image-based approaches are most suitable for intra-hour forecasting, which has competitive alternatives, such as persistence models. For example, a persistence method performed superior to machine learning methods, which made the latter ones less competitive for 15 min time granularities in the additive mode [9]. In practice, most utilities rely on persistence models for intra-hour solar forecasting [10]. Recently, utilities have started to seek more precise intra-hour forecasting models. For example, the California Independent System Operator (CAISO) has continuous interest in developing intra-hour forecasting methods that provide 5-minute forecasts for the next 2 hours [11]. Another reason is the unclear scalability of the approach, largely due to the low deployment of sky camera infrastructures. Specifically, there is limited research on the ability to aggregate sky image-based solar forecasts for individual solar sites into higher level forecasts in power systems, such as load zones and balancing areas. Finally, the higher complexity and costs of image-based deep learning models must be hurdled, compared to numerical time series-based models. As graphics processing units (GPUs) become more accessible, processing sky images with deep learning techniques for solar forecasting has become an emerging topic [12–15].

With the motivations and new techniques discussed above, sky image-based solar forecasting has attracted attention in recent years. Yet there are serious drawbacks to the research currently conducted in this area. First, most models rely on empirically-defined sky image features, such as pixel statistics, image segments, and cloud movement characteristics. For example, three statistics of sky image pixel red blue ratios, i.e., mean, standard deviation, and Rényi entropy, were used in conjunction with meteorological parameters for 1HA GHI [16]. Numerical metrics of cloud coverage are often also important features. In Chu et al. [17] and Marquez and Coimbra [18], cloud indices were extracted from sky images to predict 3MA–10MA direct normal irradiance (DNI). Cloudy pixel percentages of six grid elements originating from the sun were used as the input to artificial neural network (ANN) models, which achieved over 20% forecast skill scores [17]. Moreover, the cloud velocity was proved to have a significant impact on the DNI forecasting accuracy. Forecast root mean square error increased by 24.2% and 8.4%, respectively, with an overestimation and underestimation of the cloud velocity magnitude by 50% [19]. Although being effective in solar forecasting, the above sky image parameters require complex feature engineering and domain-specific expertise. It has been proved in various fields that machine learned features are better than hand-crafted features for different tasks [20–22]. Although being effective in solar forecasting, these hand-crafted image features may not be optimal for solar forecasting.

Deep learning models have the potential to provide a successful solution to the manual interventions required, with only simple pre-processing procedures. One of the key features of deep learning is the end-to-end learning fashion by gradient descent, which has been proven many times in various fields [23–26]. However, this end-to-end learning capability was not fully utilized in some sky image solar forecasting, where deep learning classification networks were used to extract sky image features for other forecasting regression models [12,13]. For example, solar forecasting was modeled as a classification problem in Pothineni et al. [12], where continuous irradiance was converted into binary values using a clear sky index threshold. Additionally, a convolutional neural network (CNN) was used to extract image features,

which are then fed into a linear autoregressive model and a multilayer perceptron model [13].

In addition, some deep learning models for solar forecasting have not considered state-of-the-art deep learning architectures. For example, a 3-convolutional-layer CNN was used in Kong et al. [27], which might not be deep enough to forecast solar data that involves complex patterns. Similarly in Zhao et al. [28], a feature extractor with two convolutional layers was used to map sky image features to solar irradiance. Typically, an efficient deep learning network consists of 10–20 convolutional layers for classification problems [29]. Regression tasks, with more complicated continuous outputs, require at least similarly deep architectures for efficient feature learning. When fused with other layers (e.g., dense layers), the network should be even more complex to be effective. A high risk of underfitting may exist in shallow CNNs but was not always realized in the literature. For example, a network with 2 convolutional layers and residual blocks was used in Zhang et al. [30] to extract sky image features. A network with 3 convolutional layers and a concatenated dense-layer head was found to mimic the persistent method in 2MA–20MA solar irradiance forecasting [31]. The researchers realized this could be improved by upgrading the network with recurrent units. A network with 4 convolutional layers even showed less accurate forecasts when compared to persistence [32].

Finally, some research verified deep learning models with limited data, which could not provide general and convincing results. A small *training* dataset might suffer from overfitting, especially in deep learning. However, this was not realized in some research. In Zhang et al. [30], only 6 months of data were used to train a complex deep learning network with convolutional layers, long short-term memory layers, and dense layers. The training dataset in another study consisted of only around 30 days of images [27]. A more extreme example used only 7 days of data to train CNNs [32]. A related issue is using too small of a *testing* dataset to show that the results are generalizable. For example, a 20-day dataset was used to verify the forecasting methods in four cloud conditions [33]. Similarly in Zhang et al. [30], 20 days of sky images from two months were used as the testing dataset. A 3-day testing dataset showed that the CNN provided worse forecasts than persistence in 5MA–20MA forecasting [32].

To address the challenges described above in sky image-based solar forecasting, this paper develops two deep CNN methods taking sky image sequences as input for operational intra-hour solar forecasting. The first 2-dimensional CNN method stacks image sequences in the channel dimension and the second 3-dimensional (3D) CNN method applies 3D feature learning. Both methods are trained in an end-to-end manner. Input hyperparameters are also optimized through sensitivity analyses. The contributions of this paper include: (i) developing two end-to-end sky image-based CNN configurations for operational solar forecasting, (ii) identifying the best input setup by sensitivity analyses and interpreting the machine learned image features by filtered feature maps, (iii) achieving an average forecast skill score of 20% based on 6 years of open-source data. The remainder of this paper is organized as follows. Section 2 introduces operational solar forecasting, formulates sky image-based intra-hour solar forecasting, and develops the two deep CNNs. The experimental setups, including the dataset, case studies, baseline and benchmark models, are described in Section 3. Section 4 details forecasting results and performs sensitivity analyses. Section 5 discusses forecasting limitations, future work, deployment challenges and opportunities. Section 6 concludes the paper.

2. Methodology

Both power system operators and energy market participants require timely and accurate solar forecasts. This section first introduces operational solar forecasting requirements for individual PV plants. Then, the intra-hour solar forecasting problem is formalized. Finally, the deep CNNs are developed to solve the intra-hour solar forecasting problem.

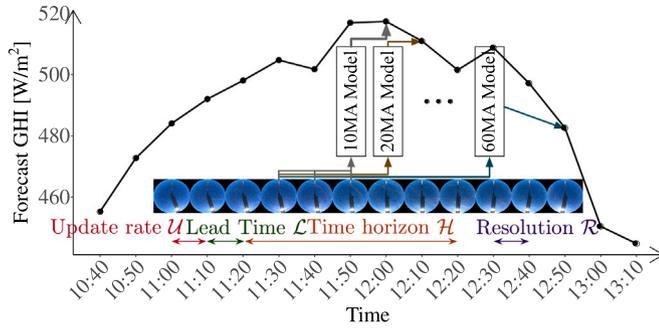


Fig. 1. Operational intra-hour solar forecasting. In this research, $U = 10$ -min, $L = 10$ -min, $H = 60$ -min, $R = 10$ -min.

2.1. Operational solar forecasting

In this study, we propose a methodology to provide operational 1HA solar forecasts, which can be directly used in daily utility operations. Operational forecasting refers to forecasting problems that fit utilities' operational needs, which are regulated by independent system operators (ISOs) or regional transmission organizations according to the electricity markets [34]. For example, in the CAISO real-time energy market, PV plant operators are required to submit intra-hour forecasts with a 65-min horizon and a 5-min resolution, which are used to commits resources in the 15-min real-time unit commitment and the 5-min real-time economic dispatch. The submission is due 10-min before every operating time and should be updated every 15-min. Operational forecasting is more complex than forecasting defined in the majority of literature, where only fixed-step ahead forecasts are generated. There are at least four time-related parameters in operational forecasting problems—lead time (L), time horizon (H), resolution (R), and update rate (U) [34]. Formulating solar forecasting in the form of operational forecasting not only enhances the value of the forecasts related to their actual application, but also bridges the gap between the forecasting community and the power systems community.

2.2. Intra-hour forecasting formulation

In this research, we seek to develop a sky image sequence-based deep learning methodology to forecast intra-hour GHI¹ in an operational forecasting manner. Multi-step ahead forecasting is achieved through six parallel models, which is demonstrated in Fig. 1. Based on our dataset (detailed in Section 3.1), forecasts provided by the six independent models span 1-hour with a 10-min lead time, a 10-min resolution, and a 10-min update rate. Each forecasting model takes a sky image sequence as its input and predicts a single future GHI value at each forecasting issue time. The sky image sequence to GHI mapping is formulated as:

$$y_{\Delta t} = F_{\Delta t}(\mathbf{X}, \mathbf{W}_{\Delta t}) \quad (1)$$

where $\mathbf{X} \in \mathbb{R}^{N \times W \times H \times D}$ is the image sequence input; $y \in \mathbb{R}^{N \times 1}$ is the actual GHI; N , W , H , D are the sample number, the image width, the image height, and the image channel size (a typical digital image has red, green, and blue channels), respectively; $\Delta t \in \{10, 20, \dots, 60\}$ is the lead time in terms of minutes; \mathbf{W} is the trainable parameter matrix in CNN models.

To perform this very short-term solar forecasting (VSTSF), optimal parameters should be obtained first at the training stage. Since VSTSF is a regression problem, we use L1 (e.g., mean absolute error) loss

¹ We target forecasting the GHI instead of PV power due to the lack of measured power data for the image location.

function to respect the overall performance of the model and reduce outlier impacts [35]:

$$J(\mathbf{W}_{\Delta t}) = \frac{1}{N} \sum_{n=1}^N |\hat{y}_{\Delta t, n} - y_{\Delta t, n}| \quad (2)$$

where n is the sample index; y and \hat{y} are actual and forecast GHI values, respectively. Then, the objective of the deep learning training is to optimize the parameters, $\mathbf{W}_{\Delta t}$, by minimizing the loss function (i.e., reducing the forecast error) in an end-to-end manner (i.e., from image sequences to GHI values). Once the optimal model parameters, $\mathbf{W}_{\Delta t}^*$, are obtained, the well-trained deep learning models are used to generate forecasts in the forecasting/testing stage:

$$\hat{y}_{\Delta t} = F_{\Delta t}^*(\mathbf{X}, \mathbf{W}_{\Delta t}^*) \quad (3)$$

where \hat{y} is the forecast GHI vector; F^* indicates the well-trained deep learning model.

2.3. Deep learning architectures

Deep learning has been widely used to process images in the computer vision application field, such as in medical image processing, automatic driving, and surveillance. Deep learning-based image processing is less commonly studied in the power and energy field, where most applications are classification problems [36–39]. In this research, we develop CNNs for sky image processing for solar forecasting, which is defined as a regression problem.

Unlike most sky image-based solar forecasting that uses shallow machine learning, this research sets up the model in an end-to-end (i.e., image sequence to GHI) manner. This means that the forecasting does not rely on complex feature engineering, such as feature extraction, selection, and reconstruction. We design two deep CNN architectures to learn the latent patterns between the GHI and sequences of images. CNN architectures are preferred over other deep learning architectures for the following reasons: (i) CNNs are more powerful than other architectures in image processing regarding their feature learning capabilities and performance, since convolutional layers are able to deal with inherent properties of images by using kernels and convolution [40,41]; (ii) image sequence-based solar forecasting using CNNs has lagged behind, which requires further investigation; (iii) other successful architectures, such as long short-term memory, have a higher level of complexity, which is beyond the scope of this study.

2.3.1. The VGG-like feature extractor

The two architectures are inspired by the popular very deep convolutional network developed by the Visual Geometry Group (VGG) [42] and optimized for our intra-hour solar forecasting problem. The reasons for selecting the VGG-like CNNs are that: (i) the VGG networks are recommended by the deep learning community for their performance in various deep regression problems (regression problems solved by deep learning) [29]; (ii) the complexity of VGG networks is lower than their counterparts, such as the ResNet²; (iii) the forecasting accuracy of the VGG networks is higher in our case studies with smaller datasets.³ The overall frameworks of the two CNNs are shown in Fig. 2. The first architecture is a 2-dimensional CNN that processes stacked images in two directions—the height and width of images (denoted as SCNN). The second architecture is a 3-dimensional CNN that directly conducts 3D tensor calculations (denoted as 3DCNN). Each architecture consists of convolutional layers, max-pooling layers, and densely-connected layers.

² The ResNet is eight times deeper than VGG nets [43].

³ We compare multiple deep learning networks, including the VGG networks, the ResNets, and the Inception Networks with two year data (2012–2013). The VGG networks outperform their counterparts. However, comparing different deep learning architectures exhaustively is beyond the scope of this paper, which is not discussed in the paper.

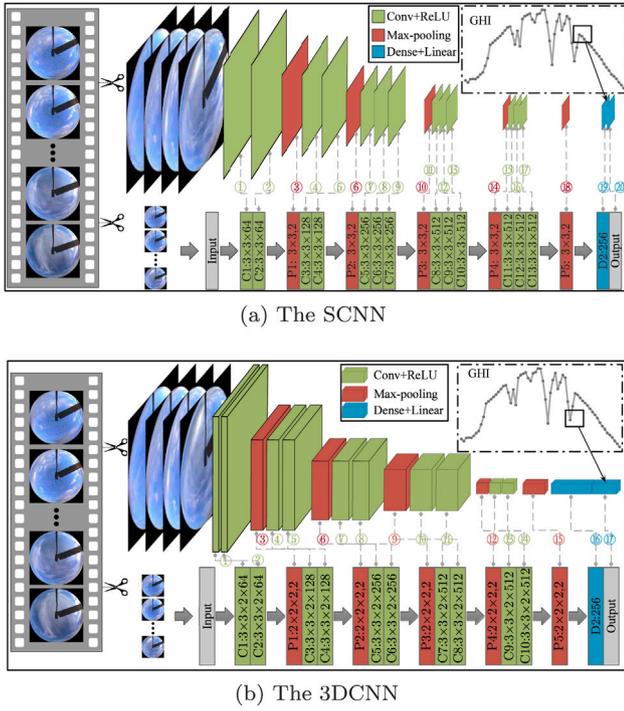


Fig. 2. Frameworks of the two developed deep learning architectures..

The general architectures of the two methods are similar. As shown in Fig. 2, the developed architectures contain 5 feature learning blocks (FLBs), each of which stacks 2 or 3 convolutional layers and a max-pooling layer, which is shown in the upper parts of Figs. 2(a) and 2(b). The two networks utilize small filters to extract more receptive fields from the input tensors. The filter number of the next FLB is doubled to extract more abstract and informative features.

Convolutional layers are the core building blocks of the two deep learning architectures. They extract features from sky images by convolving the input of each layer with its filters, which is expressed as $\mathbf{Z}^l = \mathbf{W}^l * \mathbf{X}^l + \mathbf{b}^l$. \mathbf{X} , \mathbf{W} , \mathbf{b} , and \mathbf{Z} are the input tensor, parameter tensor, bias tensor, and output tensor in the l th layer, respectively; $*$ is a convolution operator. Then, the convolution results are applied to an activation function and output feature maps. Low-level features, such as dots, lines, and curves of sky images, are learned by the first convolutional layers. In higher convolutional layers, increasingly complex features are extracted on top of previous sky image features.

Among convolutional layers, max-pooling layers pick out maximum feature values of local regions to extract representative features while reducing the dimension. The pixel window in each block is 2×2 or $2 \times 2 \times 2$ (depending on which architecture) with a stride of 2. The pooling layers introduce more translation invariance during the spatial representation learning, by sub-sampling convoluted feature maps by the non-overlapping pooling windows.

2.3.2. The SCNN

The top part of Fig. 2(a) shows the structure of SCNN models and the bottom part shows the layer hyperparameters. SCNN models take stacked images as input and output GHI forecasts. There are five feature learning blocks with increasing numbers of filters. Each block (except for the first block) contains 2–3 convolutional layers and a max-pooling layer. The convolutional and max-pooling layers are powerful in extracting informative features from the sky images, but cannot directly solve the regression problem. Therefore, fully-connected layers are used to learn the non-linear combinations of sky image features. All the inputs are transmitted to the output in fully-connected layers, as: $\mathbf{Z}^l =$

$\mathbf{W}^l \cdot \mathbf{X}^l + \mathbf{b}^l$. Two fully-connected layers are included in the last layers for regression. The first fully-connected layer has 256 neurons and the output layer has 1 neuron. The numbers of layers and neurons are optimized by the grid search. The rectified linear activation function is used in convolutional layers and the linear activation function is used in fully-connected layers. A 0.2 dropout is optimally selected [44] by the grid search and used before each fully-connected layer to avoid overfitting.

2.3.3. The 3DCNN

The 3DCNN structure is similar to the SCNNs, as shown in Fig. 2(b). In contrast to 2D CNNs that treat each depth channel separately, 3DCNN uses 3D filters to extract features. There are also 5 feature learning blocks in 3DCNN models. Each block has convolutional layers and one max-pooling layer. Due to the nonisotropic image input, the filter dimension is $3 \times 3 \times 2$. The same filter numbers and activation functions are used in 3DCNN models for fair comparisons. The same dense layers are used on top of the 3DCNN blocks.

2.3.4. Training deep learning models

A prerequisite of an accurate model is a successful training process. According to Lathuilière et al. [29], the adaptive moment optimization (Adam) is used to optimize CNN parameters that are initialized with the pre-trained ImageNet weights. The mean absolute error is used as the loss function to be minimized by updating parameters in the opposite direction to their gradients:

$$m_{i,j} = \beta_1 m_{i,j-1} + (1 - \beta_1) \nabla_{w_i} J(\mathbf{W}; \mathbf{B}_j) \quad (4)$$

$$v_{i,j} = \beta_2 v_{i,j-1} + (1 - \beta_2) \nabla_{w_i}^2 J(\mathbf{W}; \mathbf{B}_j) \quad (5)$$

$$\hat{m}_{i,j} = \frac{m_{i,j}}{1 - \beta_1^j} \quad (6)$$

$$\hat{v}_{i,j} = \frac{v_{i,j}}{1 - \beta_2^j} \quad (7)$$

$$\omega_{i,j} = \omega_{i,j-1} - \Delta \omega_{i,j} = \omega_{i,j-1} - \eta \frac{\hat{m}_{i,j}}{\sqrt{\hat{v}_{i,j} + \epsilon}} \quad (8)$$

where m and v are, respectively, exponential moving averages of first and second moments of the gradient along network parameter ω_i in the j th mini-batch, $\mathbf{B}_j = [\mathbf{X}_j, \mathbf{y}_j]$ ($m_{i,0} = v_{i,0} = 0$). $\beta_1 = 0.9$ and $\beta_2 = 0.999$ are Adam parameters controlling the decay rates of the moving averages. $\eta = 0.01$ is the initial learning rate and $\epsilon = 1 \times 10^{-8}$ is a constant to prevent zero division. In the training stage, the models are passed through the training and validation datasets 60 times, during which mini-batches with a batch size of 64 (determined based on the GPU memory and the validation accuracy) are randomly generated to shuffle the data order. Gradients are averaged over the mini-batch in each iteration to update the weight matrix, \mathbf{W} .

3. Experimental setup

3.1. Dataset description and pre-processing

Following the suggestion of using open-source datasets to practice solar forecasting [34], this research conducts all the experiments on a publicly available dataset—the National Renewable Energy Laboratory (NREL) solar radiation research laboratory (SRRL) dataset. The SRRL dataset is one of the largest publicly available datasets with both total sky images and meteorological measurements. The data has been collected since 1981 at the South Table Mountain Campus of NREL (longitude: 105.18° W, latitude 39.74° N, elevation 1,828.2 m). More information about the dataset can be found in Stoffel and Andreas [45]. The data can be accessed through the OpenSolar package [46].

Two types of data, namely, the total sky images and numerical meteorological measurements, are downloaded from the NREL SRRL

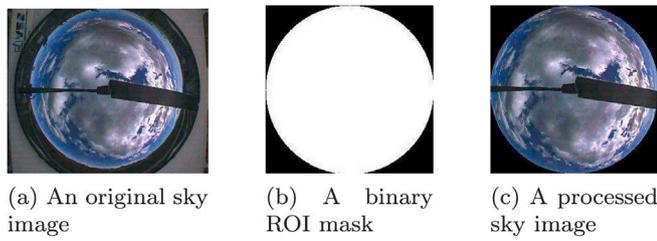


Fig. 3. Sky image pre-processing.

database. The sky images taken by a Yankee Total Sky Imager (TSI-800) every 10-min are selected for the case studies. Six years of sky images are downloaded and pre-processed. Every image contains 352×288 pixels. The developed networks require minimal preprocessing due to their end-to-end learning capability. The only preprocessing is using a binary mask to circularly crop the images to avoid the hazy sky and obstacle presence. Every image with a region-of-interest (ROI) has 256×256 pixels. Fig. 3 demonstrates this only image pre-processing applied in this research.

Meteorological measurements are also collected to guide the network training and to build the benchmark models. Numerical measurements have a 1-min resolution, which are averaged to 10-min to keep consistency with sky images. Numerical parameters include temperature, relative humidity, wind speed, GHI, DNI, diffuse horizontal irradiance (DHI), and atmospheric pressure. Irradiance parameters are normalized by their corresponding clear sky irradiance, while other parameters are normalized by their maximum values. Therefore, the forecasting target parameter is the clear sky index:

$$y := \text{CSI} = \frac{\text{GHI}}{\text{CSGHI}} \quad (9)$$

where CSGHI is the Ineichen and Perez clear sky GHI simulated in the pvlb Python package [47]. Fig. 4 shows the GHI, CSGHI, CSI, and their coincident sky images under various weather conditions.

3.2. Case studies

There are 155,644 data points after filtering out nighttime data points and aligning images with numerical data. To ensure the successful training and convincing verification, six years (i.e., from 2012-01-01 to 2017-12-31) of data are used and divided into a training dataset, a validation dataset, and a testing dataset. The training dataset consists of the first three years (i.e., 2012-01-01 through 2014-12-31) data. The following year of data (i.e., from 2015-01-01 to 2015-12-31) are used for validation, and the last two years data are used for testing (i.e., from 2016-01-01 to 2017-12-31). The validation dataset is a hold-out sample set used to tune model hyperparameters. All the results in Section 4 are based on the testing dataset.

In addition to different machine/deep learning methods, sensitivity analyses are performed to investigate the impact of the image sequence length, image resolution, and lead time on the forecasting accuracy. Four image sequence lengths, i.e., $2^0, 2^1, 2^2, 2^3$ image(s), three image resolutions, i.e., $256 \times 256, 128 \times 128, 64 \times 64$, and six lead times, i.e., 10MA–60MA, are included in the case studies. Therefore, a total of 144 models are built to cover all the scenarios of the two deep learning-based forecasting methods (4 image lengths \times 3 image resolutions \times 6 lead times \times 2 methods). Sample sizes of the case studies differ due to the different image sequence lengths and lead time scenarios. Table 1 lists the sample sizes of 24 experiments corresponding to one of the six model-resolution combinations. It is natural that taking more images as the model input or forecasting a longer lead time reduces the sample size due to the data alignment. Using such settings corresponds to more realistic compromises in real-world forecasting practice.

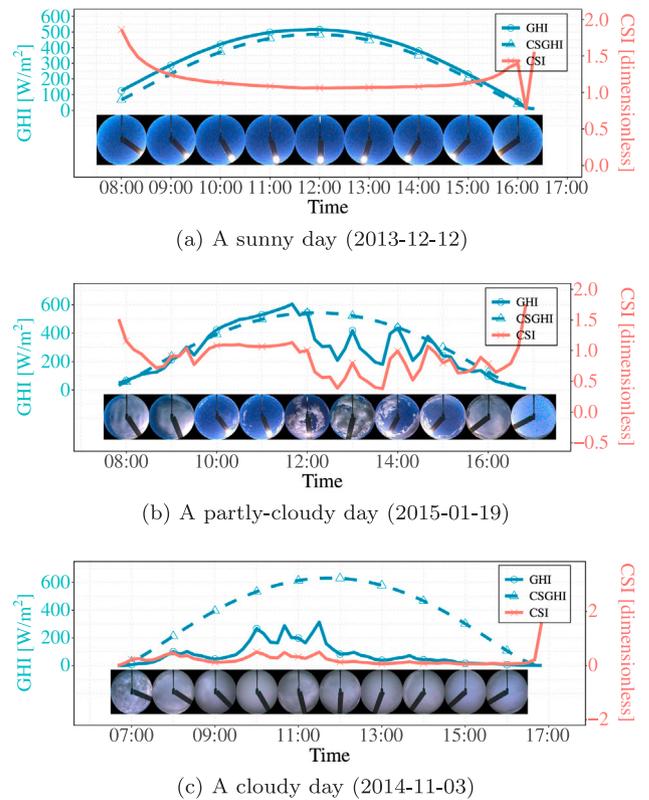


Fig. 4. GHI, CSGHI, CSI time series, and their corresponding sky image sequence under three weather conditions. The color indicates the three irradiance features. Sky images with a 10-min resolution are used in the case studies, but only hourly images are included in this figure due to the limited space. The numerical parameters are also averaged to 10-min and shown in the figure. The hourly values are indicated by the points.

Table 1
Sample sizes in different image sequence length and lead time scenarios.

Image No.	\mathcal{L} [min]	Sample size		
		Training	Validation	Testing
1	10	76,708	26,271	52,665
	20	75,882	25,989	52,109
	30	74,820	25,624	51,381
	40	73,748	25,258	50,653
	50	72,681	24,895	49,925
	60	71,612	24,531	49,196
2	10	75,482	25,827	51,795
	20	74,659	25,545	51,241
	30	73,593	25,179	50,516
	40	72,524	24,815	49,791
	50	71,459	24,453	49,065
	60	70,393	24,091	48,339
4	10	73,048	24,940	50,069
	20	72,222	24,658	49,520
	30	71,161	24,295	48,800
	40	70,097	23,934	48,080
	50	69,040	23,576	47,357
	60	67,978	23,218	46,632
8	10	68,239	23,200	46,672
	20	67,418	22,923	46,126
	30	66,368	22,568	45,410
	40	65,314	22,215	44,691
	50	64,262	21,863	43,973
	60	63,207	21,509	43,253

Table 2
Machine learning benchmark models.

Model	Function/Algorithm	Hyperparameter
ANN1	Momentum back-propagation	max_epoch=1,000, momentum=0.9, lr=0.01
ANN2	Standard back-propagation	max_epoch=1,000, lr=0.01
GBM1	Squared loss	ntrees=1,000, max_depth=20, bag_frac=0.5, lr=0.01
GBM2	Laplace loss	ntrees=1,000, max_depth=20, bag_frac=0.5, lr=0.01
GBM3	T-distribution loss	ntrees=1,000, max_depth=20, bag_frac=0.5, DF=4, lr=0.01
RF	Classification & regression trees	ntrees=1,000, mtry=5

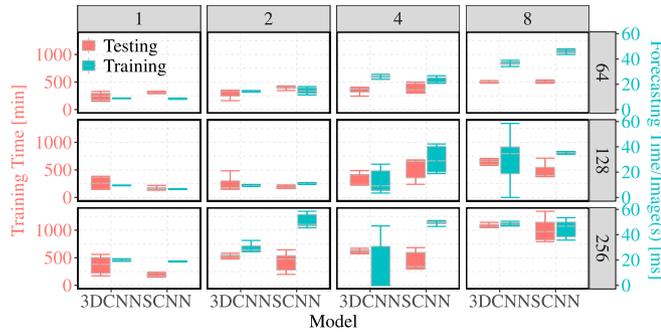


Fig. 5. Deep learning computational time. The red color indicates training time by minute, the blue color indicates forecasting time per data point by millisecond. The statistics are grouped by lead time in columns and by resolution in rows.

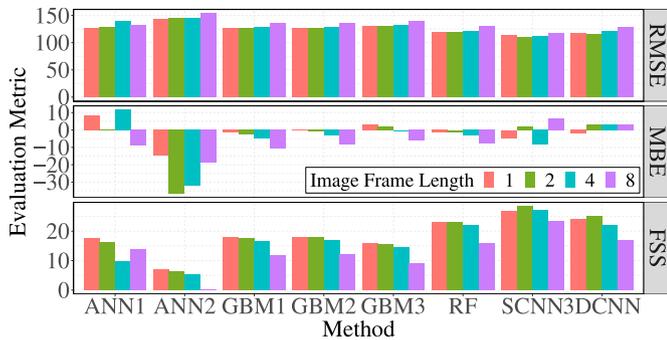


Fig. 6. Averages of forecast RMSEs [W/m²], MBEs [W/m²], and FSSs [%] over different lead times. The colors represent the four input lengths.

3.3. Baseline and benchmark models

A collection of six machine learning methods, in conjunction with the persistence of cloudiness (PoC) method, are used as benchmarks and baseline. The machine learning methods are selected from a larger volume of methods based on their performance in the 10-fold cross-validation study. The selected machine learning benchmarks include two ANNs, three gradient boosting machine methods (GBMs), and a random forest (RF) method. The hyperparameters and parameters of the benchmark models are listed in Table 2. Additional detailed parameter information can be referred in Refs. [48–52]. The inputs to machine learning models include GHI, DNI, DHI, clear sky GHI, clear sky DNI, clear sky DHI, dry bulb temperature, wind chill temperature, relative humidity, wind speed, peak wind speed, pressure in the current and past hour(s). The same data partition is applied to the benchmark modeling. Hence, the total data lengths are slightly longer than the image-involved models. The PoC method assumes a constant CSI within the forecasting lead time [53,54].

4. Results

Case studies are conducted on high performance computing (HPC) GPU nodes in the Texas Advanced Computing Center (TACC)⁴ Maverick2 system at the University of Texas System. Each TACC node contains 16 Intel(R) Xeon(R) CPUs and 4 Nvidia GTX 1080-TI GPUs. The benchmark case studies are conducted on Ganymede HPC nodes at the University of Texas at Dallas, which has 40 Intel(R) Xeon(R) CPUs. The SolarNet and benchmark models are implemented using the Keras library with Tensorflow backend in Python 3.7.0 and the caret package in R. Detailed experiment implementations can be found in our SolarNet GitHub repository. The computational time of deep learning training and forecasting is shown in Fig. 5. Three recommended evaluation metrics, the root mean square error (RMSE), the RMSE-based forecast skill score (FSS), and the mean bias error (MBE), are used to evaluate the overall performance of models [55].

4.1. Overall performance

The overall performance of intra-hour solar forecasting models is shown by the averages of each metric over different lead times in Fig. 6. Generally, all of the machine learning methods have superior performance compared to the PoC method at such short lead time forecasting. This is evidenced by the positive FSSs. Compared to shallow machine learning, the two deep learning methods further improve forecasting accuracy. The SCNN model has the highest 28.49% FSS among all models with a competitive MBE. Tables 3–5 break down the averages into metrics based on lead times and input sequence lengths. First, in most cases, the machine learning models have positive FSSs, although the PoC method is often considered sufficiently accurate at these timescales. Therefore, it is suggested to include meteorological parameters in intra-hour solar forecasting if they are available. More importantly, the SCNN models yield the most accurate solar forecasts, which encourages total sky imagers (TSIs) to be used as a complementary or even an alternative solar forecasting sensor. This will be especially beneficial to distributed solar systems, considering the low equipment and installation cost of the TSIs.

The forecast RMSEs increase with the lead time as expected. In shallow machine learning, this can be explained by the decreased autocorrelations within the meteorological parameters, especially in the GHI series itself. Similarly in deep learning, autocorrelations of the image sequence decay as the time lag increases. The SCNN models generate more accurate forecasts than shallow learning models in all cases, while the 3DCNN models perform better in most cases. The average FSS of the SCNN model is 26.45%. Considering the challenge of improving forecasting accuracy in such short lead times, the improvements provided by the developed deep learning models are significant. When compared to RF, the best shallow learning method, the forecast RMSEs are reduced by around 7%, which is still considerable.⁵

By comparing the two deep learning models, it is also observed that the SCNN performs better than the 3DCNN in these solar forecasting tasks. The first reason is that 3DCNNs are computationally inefficient

⁴ <http://www.tacc.utexas.edu>.

⁵ Improving accuracy by even 1% is a great achievement in deep learning [56].

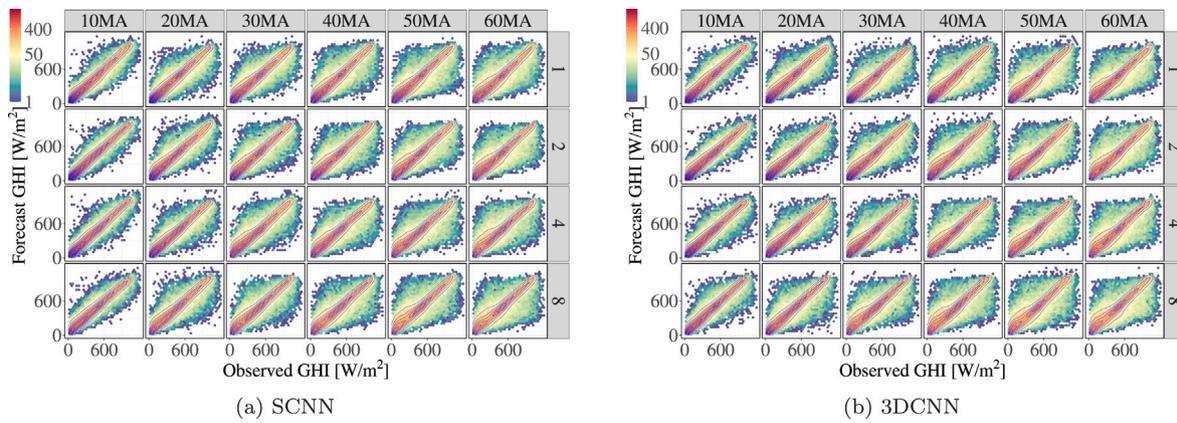


Fig. 7. The joint distributions of forecast versus observed GHI. For a higher contrast, the color scheme is based on the logarithm of scatter point frequency. The plots group distributions by lead times in columns and image sequence lengths in rows.

Table 3
Forecast RMSEs [W/m^2] of the best SCNN, 3DCNN, and benchmark models.

\mathcal{L} [min]	Sequence Length	ANN1	ANN2	GBM1	GBM2	GBM3	RF	PoC	SCNN	3DCNN
10	1	94.52	88.92	87.23	87.16	88.89	85.36	90.71	79.90	81.82
	2	85.28	88.71	87.80	87.67	89.53	83.66	90.77	71.30	80.28
	4	84.51	90.97	88.91	88.79	91.04	85.39	91.68	73.76	87.38
	8	94.49	147.89	93.07	92.98	95.53	90.26	94.11	81.67	92.04
20	1	111.53	115.71	113.70	113.77	116.50	108.58	122.85	103.53	102.87
	2	132.79	122.81	114.09	114.05	117.17	108.98	123.63	98.53	101.73
	4	110.47	121.09	115.28	115.13	118.99	109.60	123.21	98.27	106.86
	8	121.12	132.17	120.96	120.84	125.45	117.33	125.15	107.35	116.69
30	1	123.42	207.87	127.22	127.16	130.30	121.01	144.66	111.07	118.16
	2	119.41	143.39	127.52	127.35	130.74	119.59	145.67	109.33	115.73
	4	130.69	126.96	128.90	128.60	132.64	121.63	147.55	111.72	121.11
	8	130.12	137.62	135.70	135.43	140.66	130.52	140.67	113.69	127.89
40	1	132.24	144.59	136.64	136.36	140.00	127.34	167.42	118.98	127.24
	2	142.73	148.00	136.92	136.56	140.33	127.54	168.50	119.35	127.58
	4	157.26	158.46	138.54	138.09	142.12	129.70	170.83	121.48	126.67
	8	144.06	182.62	146.56	146.07	151.32	140.21	151.59	126.66	137.21
50	1	138.85	150.71	144.22	143.58	147.36	132.75	192.94	130.67	136.38
	2	148.82	166.43	144.63	144.01	147.77	133.32	194.18	127.49	129.49
	4	136.49	148.22	146.32	145.65	149.41	135.62	196.84	131.29	137.43
	8	148.34	160.51	155.61	154.82	159.82	146.86	161.61	136.10	146.43
60	1	162.98	153.64	150.92	150.01	153.91	136.96	221.82	133.12	135.49
	2	145.11	198.57	151.37	150.52	154.45	137.75	223.20	135.43	137.05
	4	216.87	228.69	153.17	151.99	156.00	140.29	226.24	137.55	142.77
	8	157.38	162.25	164.01	162.69	167.57	152.97	183.85	142.95	148.23

Note: The maximum GHI in the testing dataset is $1,268.393 \text{ W/m}^2$, while the maximum GHI in the entire dataset is $1,319.328 \text{ W/m}^2$. Due to the data alignment, PoC forecast RMSE differs with sequence length for the same lead-time forecasting.

to train. All the matrix operations in 3DCNN models are 3-dimensional, resulting in more parameters than 2-dimensional CNNs with similar numbers of layers. The second reason is the nonisotropic nature of the 3D sky image input. Different from other 3D images with three isotropic dimensions (i.e., computed tomography images), sky images in 3DCNNs consist of temporally changing pixels in the third dimension. This will conflict with the isotropic kernels used in the 3DCNN models. Nevertheless, their competitive performance motivates further development in the following research.

4.2. Detailed comparison

Beyond the metrics of overall evaluation, we also analyze the forecasting results intuitively. The first approach is the joint distribution of observations and forecasts, which communicates the model performance in a time-independent manner. The goodness of forecasts is usually assessed by the dispersion of scatter points along the identity line [55]. The joint distributions of deep learning forecasts vs. observations and benchmark forecasts vs. observations are depicted in Figs. 7

and 8, respectively. Fig. 7 separates distributions of two deep learning models in two sub-figures, while Fig. 8 groups the distributions by input length in four sub-figures. The SCNN models show the best joint distributions. First, the SCNN distributions have the best concentration along the diagonal. Second, the SCNN distributions have more balanced probabilities on both sides of the identity line.

From visual inspection, the 3DCNN models produce the second best distributions, followed by the RF models. The qualitative analysis is consistent with the evaluation metrics. Joint distributions of the 3DCNN and RF models are more dispersed than those of the SCNN. For instance, the 3DCNN with 8 images generates more underpredictions than its SCNN counterpart in 10MA forecasting. This is reflected by more points in the bottom right part of the associated facet in Fig. 7. This is more apparent in the RF distributions, where the density contours are skewed from the diagonal. The joint distributions also explain why other models perform worse. For example, the two ANN methods fail to capture the down ramps corresponding to the cloudy conditions, as revealed by the multimodal patterns in the joint distributions. GBM models place too much emphasis on the small GHI values to avoid

Table 4
Forecast MBEs [W/m^2] of the best SCNN, 3DCNN, and benchmark models.

\mathcal{L} [min]	Image No.	ANN1	ANN2	GBM1	GBM2	GBM3	RF	PoC	SCNN	3DCNN
10	1	23.64	-5.39	-0.38	-0.65	-2.02	-1.32	4.22	-1.37	19.91
	2	6.74	-6.24	-0.79	-1.13	-2.48	-0.03	4.22	1.29	6.08
	4	1.13	-12.04	-1.55	-1.91	-3.39	0.07	3.18	10.35	2.31
	8	13.12	56.54	-3.17	-3.49	-4.69	-2.02	2.52	9.15	1.94
20	1	-7.41	9.86	-0.49	-0.41	0.15	-1.10	10.12	-13.18	10.01
	2	40.12	-28.42	-1.43	-1.13	-0.48	-0.44	10.33	9.22	-1.32
	4	5.00	-19.33	-3.03	-2.56	-2.30	-0.28	8.13	-5.86	1.09
	8	-6.63	-20.42	-6.22	-5.79	-4.85	-3.17	5.42	10.35	-4.90
30	1	-2.89	-77.21	-0.96	0.42	2.79	-2.24	17.11	1.14	-19.49
	2	-6.21	-38.40	-2.13	-0.48	2.13	-0.76	17.43	4.26	-2.66
	4	21.58	-10.58	-4.48	-2.62	-0.76	-1.59	17.88	-49.17	0.29
	8	-11.88	-23.78	-9.34	-7.26	-5.54	-5.78	8.65	6.66	6.30
40	1	-17.61	28.96	-1.23	1.08	4.79	-0.17	24.81	3.63	-1.61
	2	-29.19	-36.05	-2.70	-0.23	3.82	-0.76	25.19	-2.66	10.60
	4	-36.37	-30.31	-5.68	-3.11	0.62	-2.80	25.97	-2.92	11.70
	8	2.59	-56.42	-12.29	-9.06	-5.91	-8.21	12.59	-4.29	8.85
50	1	16.25	-28.99	-1.70	0.84	5.67	-1.02	32.85	-11.75	-14.07
	2	-8.79	-44.02	-3.49	-0.66	4.56	-2.05	33.31	8.30	2.79
	4	-1.76	-29.78	-7.05	-3.85	0.93	-6.01	34.28	-6.45	-1.08
	8	-20.52	-40.17	-15.35	-11.51	-6.71	-12.26	18.10	19.58	7.41
60	1	36.68	-15.45	-2.30	0.65	6.01	-1.41	41.11	-6.18	-6.86
	2	-1.62	-66.35	-4.27	-0.85	4.80	-2.72	41.65	-8.14	4.38
	4	80.22	-89.03	-8.65	-4.64	1.50	-6.99	42.79	2.84	4.44
	8	-29.40	-27.53	-18.69	-13.94	-7.10	-15.20	28.87	-1.23	0.30

Table 5
Forecast skill scores [%] of the best SCNN, 3DCNN, and benchmark models.

\mathcal{L} [min]	Image No.	ANN1	ANN2	GBM1	GBM2	GBM3	RF	SCNN	3DCNN
10	1	-4.20	1.97	3.84	3.91	2.01	5.90	11.92	9.80
	2	6.05	2.27	3.27	3.42	1.37	7.83	21.45	11.56
	4	7.82	0.77	3.02	3.15	0.70	6.86	19.54	4.69
	8	-0.40	-57.15	1.11	1.20	-1.51	4.09	13.21	2.20
20	1	9.21	5.81	7.45	7.39	5.17	11.62	15.73	16.26
	2	-7.41	0.66	7.72	7.75	5.23	11.85	20.30	17.71
	4	10.34	1.72	6.44	6.56	3.43	11.05	20.24	13.27
	8	3.22	-5.61	3.35	3.44	-0.24	6.25	14.22	6.76
30	1	14.68	-43.70	12.06	12.10	9.93	16.35	23.22	18.32
	2	18.03	1.57	12.46	12.58	10.25	17.90	24.94	20.55
	4	11.43	13.95	12.64	12.84	10.11	17.57	24.28	17.92
	8	7.50	2.17	3.53	3.73	0.01	7.22	19.18	9.09
40	1	21.01	13.64	18.38	18.55	16.38	23.94	28.93	24.00
	2	15.29	12.17	18.74	18.96	16.72	24.31	29.17	24.29
	4	7.94	7.24	18.90	19.17	16.81	24.08	28.89	25.85
	8	4.97	-20.47	3.32	3.64	0.18	7.51	16.44	9.49
50	1	28.03	21.89	25.25	25.58	23.62	31.20	32.28	29.32
	2	23.36	14.29	25.52	25.84	23.90	31.34	34.35	33.31
	4	30.66	24.70	25.67	26.01	24.10	31.10	33.30	30.18
	8	8.21	0.68	3.71	4.20	1.11	9.13	15.79	9.39
60	1	26.53	30.74	31.96	32.37	30.61	38.26	39.99	38.92
	2	34.99	11.03	32.18	32.56	30.80	38.28	39.32	38.60
	4	4.14	-1.08	32.30	32.82	31.05	37.99	39.20	36.89
	8	14.40	11.75	10.79	11.51	8.86	16.80	22.24	19.37

large biases, which results in underpredictions for small GHI values and truncation of the forecasted GHI distributions.

Another way to interpret forecast quality is by plotting actual and forecast GHI time series. Fig. 9 exemplifies time series of three days consisting of sunny, partly cloudy, and cloudy conditions. To investigate the deep learning failure mode, testing days are ranked by forecast RMSE. A day when the SCNN model performs unsatisfactorily is selected from the worst 10 days and shown in Fig. 9(c). In sunny weather conditions (Fig. 9(a) and some time of Fig. 9(b)), the SCNN forecasts are extremely accurate, as illustrated by their overlaps with GHI actuals. In partly cloudy conditions (most time of Fig. 9(b) and

some time of Fig. 9(c)), the SCNN model is robust to the small ramps caused by the incoming clouds. However, the SCNN models, like all the forecasting models, have forecast errors, most of which occur in inclement weather conditions. Fig. 9(c) exemplifies a bad-performing day. It is observed that the SCNN performance is still good during partly cloudy hours. However, from 14:00 to 16:00 when clouds are developing, the SCNN model overpredicts GHI and yields large errors. After 16:00, the SCNN model realizes this is a cloud development period and quickly adjusts itself to the cloudy condition. Other large errors that happen less often are due to sky images with low quality (to be further discussed in Section 5).

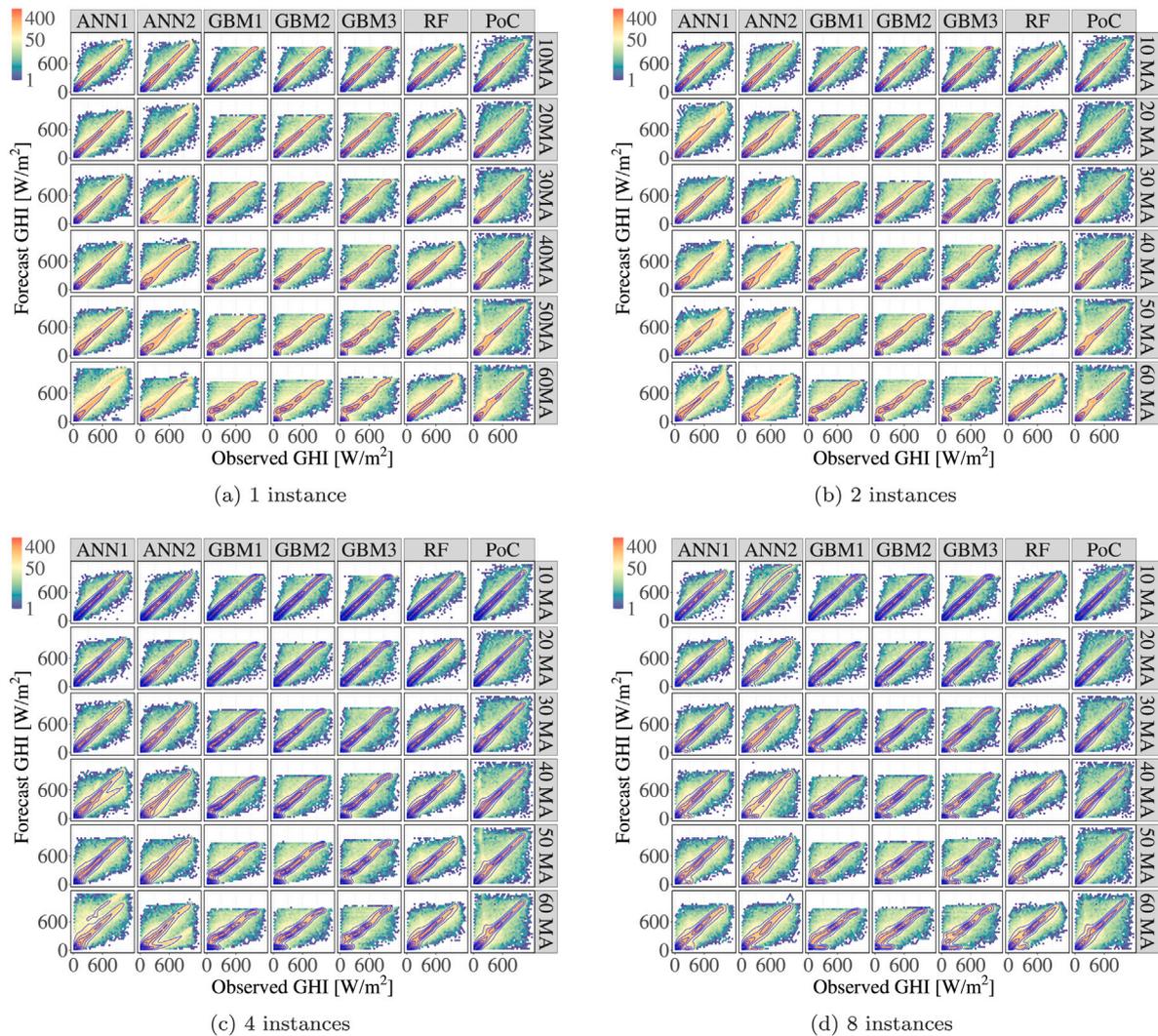


Fig. 8. Joint distributions of benchmark forecast GHI versus observed GHI. For a higher contrast, the color scheme is based on the logarithm of scatter point frequency.

4.3. Sensitivity analysis

Image resolution has a direct impact on deep learning models, which differs with different tasks, models, and experiment setups. Training with low-resolution images may suffer from missing details. On the other hand, it will take longer to train models with higher resolution images, and may also get worse results due to the increased model complexity. Like training more complex models, the issue of increasing image resolution not necessarily improving forecasting accuracy can perhaps be due to not just the risk of overfitting but also the increased complexity of the model parameter optimization problem. This research conducts experiments with the same setups except for image resolutions. Three different image resolutions are tested, i.e., 256×256 , 128×128 , and 64×64 . Fig. 10 compares the three resolutions in groups of six lead times. Images with a 128×128 resolution yield the most accurate forecasts in almost all cases since the triangle points are always the lowest point in each polygon. The 256×256 images contain redundant pixels that are not critical to solar forecasting. This is possibly due to that deep learning models learn the cloud coverage features from sky images for solar forecasting, which does not need that much detail (e.g., accurate edge and shape of the clouds). However, 64×64 images lose too much detail. Therefore, it is recommended to use 128×128 sky images for similar solar forecasting

tasks. But it is also strongly suggested to perform image resolution analysis for a specific model and experiment setup.

The length of the sequence of images⁶ is another hyperparameter that affects forecast accuracy. An image sequence consisting of more sky images is able to capture more cloud movements. But sky images of increasing lag may have weaker correlations with the current and future images. We investigate the effect of image sequence length on forecast accuracy. Four different image lengths, i.e., 2^0 , 2^1 , 2^2 , 2^3 , are included in the case studies. Fig. 11 groups lead times in six polygons and compares the three image lengths within the same group. It is found that using 2 images (10-min temporal resolution) is the best choice for our intra-hour solar forecasting. Adding more sky images will probably increase the model complexity, therefore deterring the model to learn the most effective latent patterns. This is similar to the phenomenon of using images with higher resolutions. However, using a single image does not represent the local cloud dynamics.

We have shown examples through time series plots of how the deep learning models have distinct performance in different weather

⁶ Two similar terms are used in the paper: image sequence length and input length. The first term means the length of input to image-involved models, while the latter term indicates either/both image sequence length or/and numerical measurement length.

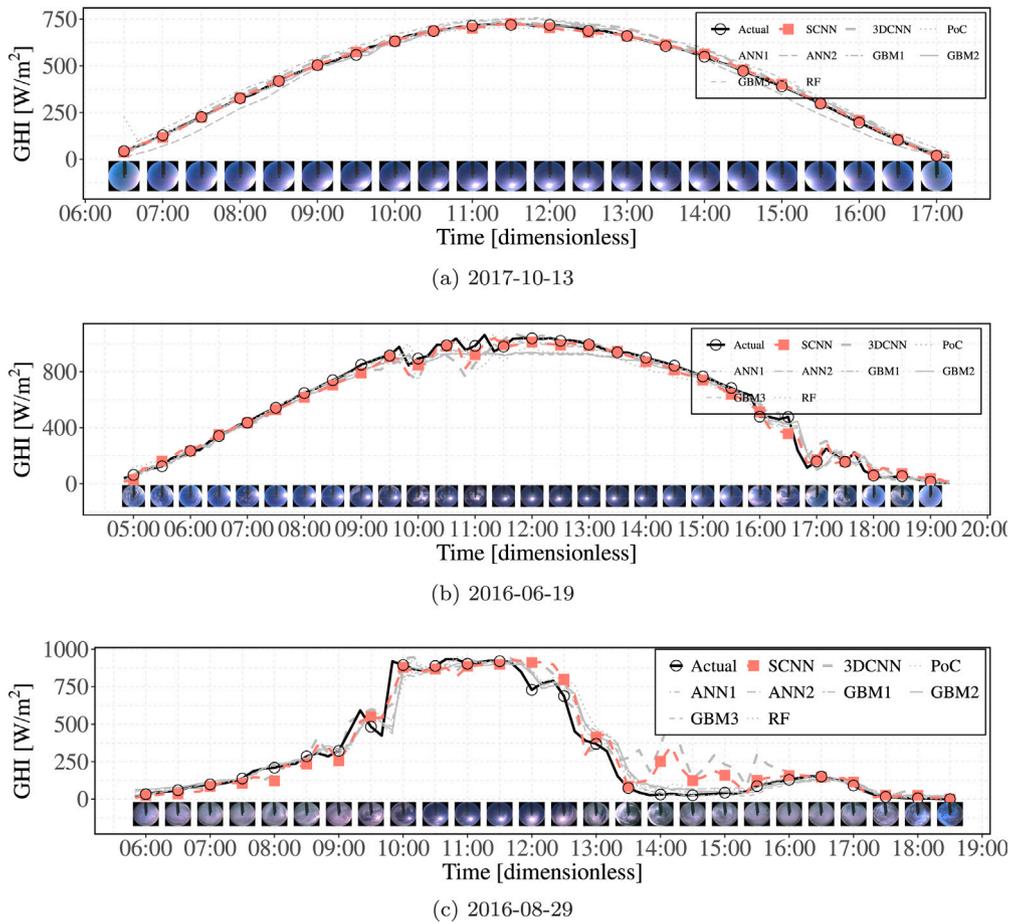


Fig. 9. Forecast and actual GHI time series. Both the numerical and image time series have a 10-min resolution. For better illustration, points and images are shown every half-hour. A typical bad-performing day (2016-08-29) is selected intentionally to investigate the failure mode of deep learning models.

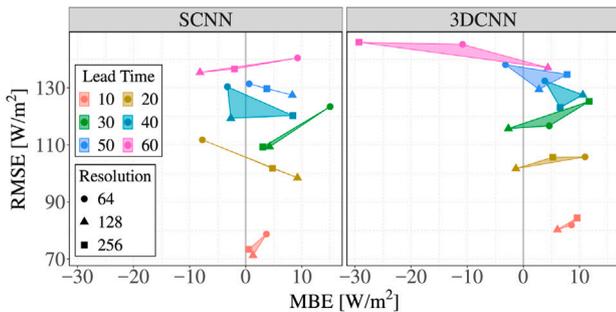


Fig. 10. Sensitivity analyses on image resolutions. The results are based on models trained with 2 images. Points with smaller RMSEs in each polygon represent models with better image resolution options.

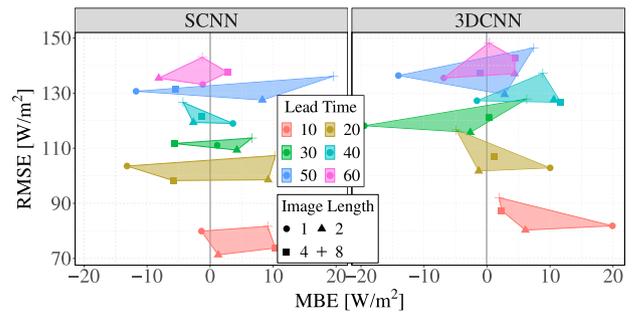


Fig. 11. Sensitivity analyses on image lengths. The results are based on models trained with 128×128 images. Points with smaller RMSEs in each polygon represent models with better image sequence length options.

Table 6
Forecast errors of the best SCNN and 3DCNN models by CSI quartile groups.

	SCNN		3DCNN	
	RMSE [W/m ²]	MBE [W/m ²]	RMSE [W/m ²]	MBE [W/m ²]
Q1	85.57	33.77	108.00	66.65
Q2	88.80	15.68	87.42	14.09
Q3	44.45	-17.71	54.72	-22.52
Q4	57.35	-25.32	69.31	-31.36

conditions. Error characteristics in Fig. 12 and Table 6 confirm the finding statistically. Groups Q1–Q4 are divided based on CSI quartiles, which means Q1 and Q2 contain more cloudy or partly cloudy hours, while Q3 and Q4 contain more sunny hours. Such categorization can result in fairer comparisons than using other CSI thresholds since the sample sizes are identical. It is shown from the peak of distributions that the deep learning models provide better forecasts for higher CSI hours, which is also proven by the smaller RMSEs in Q3 and Q4 groups in Table 6. There is a larger probability of overprediction for lower CSI, as indicated by the right tails of distributions and positive

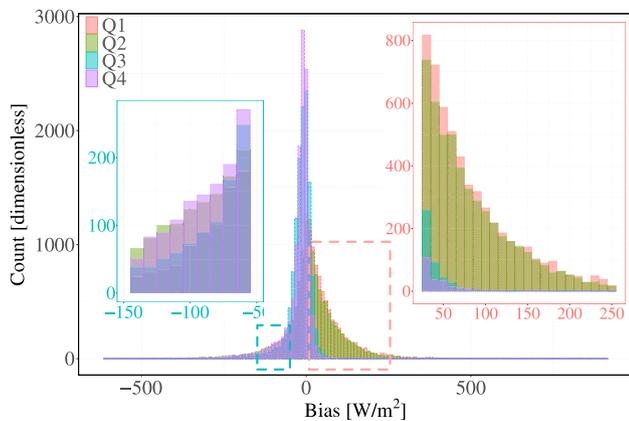


Fig. 12. Error distributions of the SCNN model in four CSI groups. Q1–Q4 divided by CSI quartiles. Q1 and Q2 contain more cloudy or partly cloudy hours, while Q3 and Q4 contain more sunny hours. Forecasts of the 10MA SCNN model with two 128×128 images are used for demonstration.

MBEs. This stimulates the bias correction in post-processing and separate forecasting tasks according to weather conditions in our future research.

In the past, CNNs were considered as a black box method with low transparency, until the recent development of deep learning interpretability visualizations. The feature map provides insight into the internal representation that deep learning has at a given point within the model. A feature map is the output of one filter applied to its input, i.e., sky images or previous feature maps depending on the depth. In this research, feature maps are used to interpret what features are extracted by deep learning models from input sky images and how these features are developed across different layers of the models. Fig. 13(a) shows the filtered feature maps convolved by all the 64 filters. It is observed that deep learning models learn to extract various features from sky images by the first convolutional layer. For example, clouds are represented by different textures in most feature maps, such as the feature maps of Filter21 (Row 2, Column 5) and Filter32 (Row 2, Column 16). Some filters focus on the background sky (e.g., Row 3, Column 5 and), while others seek the position of the sun (e.g., Row 2, Column 1). The binary ROI mask and the solar tracker are also processed by some filters, such as Filter33 (Row 3, Column 1) and Filter62 (Row 4, Column 14). Fig. 13(b) visualizes the latent feature learning process across the deep learning layers, where a representative feature map is selected from each intermediate convolutional layer. The feature maps from the first few layers (e.g., layers in Block 1) retain most of the information in the sky image. In deeper layers, feature maps look less like the original sky image and more like an abstract representation, since they encode useful features more closely related to the GHI forecasts.

5. Discussion

The developed sky image-based deep learning methods show promise for intra-hour solar forecasting. But this kind of forecasting technique is still at an early stage of widespread deployment. This section discusses the limitations of the present method, opportunities for deployment, and future work.

This research focuses on CNN methods for sky image-based intra-hour solar forecasting. Although showing superior accuracy than state-of-the-art benchmarks, the methods are found to have distinct performance under different weather conditions. First, it is worthwhile to train an individual model for each weather condition. Second, we will explore fusing sky images with other meteorological parameters through the multi-head neural networks in future work. Third,

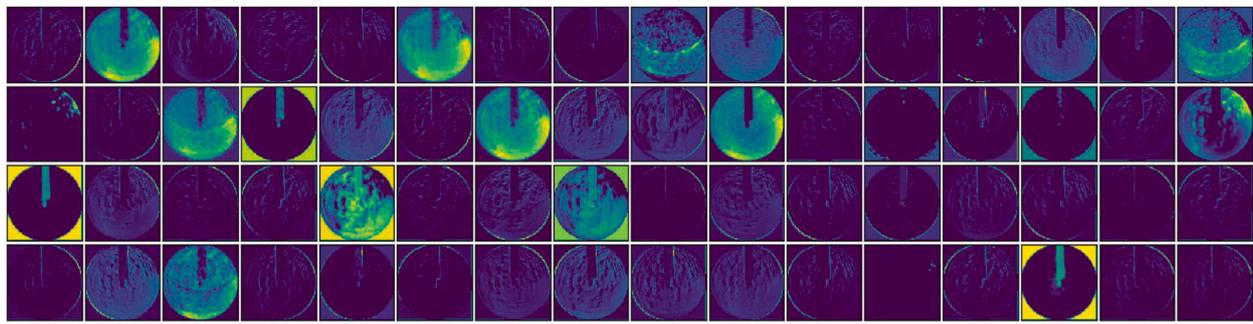
other deep learning architectures, such as recurrent neural networks, will also be investigated to process sky image sequences in future research. Finally, the uncertainty associated with the intra-hour forecasts will be quantified through probabilistic forecasting (e.g., scenario generation [57]) in future research.

Like any data-driven research, sky image-based deep learning solar forecasting requires plentiful, high-quality, and consistent data. The deep learning models are relatively computationally inefficient to train, which requires large amounts of training data. It is suggested to train/validate sky image-based forecasting models with at least 2 years' data and test with another year of data. However, it is also interesting to investigate the transferability of forecasting models trained at data-rich locations to other sites. The image quality is also critical, which can be demonstrated by Fig. 14. Low-quality images could introduce significant bias. Therefore, it is important to calibrate the sky cameras and quality control images. The following camera and lens parameters should be calibrated and kept consistent, including the geometric alignment, exposure time, focal length, bit depth, and color balance. Quality checks should be performed to flag missing and defective images. It is also better to use a solar tracker to block the direct radiation and to avoid lens flares [58,59]. Similar to the input sky images, the quality of CSI is also critical. We observed that CSI in this research could be larger than 1, which is due to the larger GHI values than the clear sky GHI values. Please note that the same CSI series is used in the deep learning models and the benchmarks. Therefore, we consider the comparisons conducted in this research are fair. However, it would be interesting to investigate different clear sky models [60] and their impacts on the forecasting models in future work.

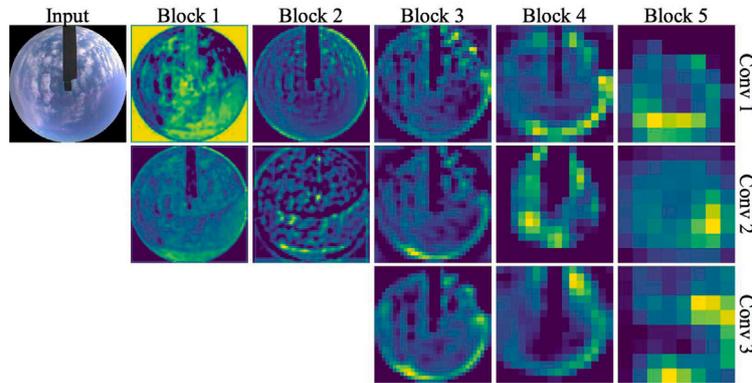
Although it has been proven that sky cameras could help power systems from different perspectives, the deployment of sky cameras is still at an early stage. Most solar plant operators rely only on meteorological measurements or even persistence methods for intra-hour forecasting. It is strongly suggested to add sky cameras to the existing weather stations of solar plants to provide another source of data. More importantly, compared to utility-scale solar plants where sky cameras serve as complementary weather devices, sky cameras could play a more critical role in behind-the-meter PV systems. This research has shown that by only using sky images, deep learning is able to provide more accurate intra-hour forecasts than commonly utilized machine learning techniques. Considering the low equipment costs [58,61], sky cameras could potentially be an alternative to meteorological devices for in situ weather measurement. 5G and cloud computing technologies make sky image-based behind-the-meter solar forecasting potentially affordable/valuable for both customers and utilities.

6. Conclusion

Accurate solar forecasting is critical to power system operations. There is a growing need to facilitate intra-hour solar forecasting. This research bridges the gaps in sky image-based operational intra-hour solar forecasting by developing two deep convolutional neural networks (CNNs) tailored for global horizontal irradiance prediction. Both CNN configurations use state-of-the-art deep learning techniques and require only sky images as input. The first CNN method processes stacked sky images in 2-dimensional convolution operations and the second CNN method directly handles sky image sequence by 3-dimensional convolution operations. Numerical experiments on 6 years' data show that the two methods outperform the persistence of cloudiness (PoC) baseline and state-of-the-art machine learning benchmarks. The average forecast skill score of the two CNNs is around 20% for intra-hour solar forecasting and the best model (i.e., SCNN) has an average forecast skill score of 28.49% over PoC. The developed CNN models are sensitive to weather conditions. They generally perform well in sunny and partially cloudy conditions. The CNN models can quickly adjust after underperforming in the few first instances of cloudy conditions. Forecasting quality is affected by both image resolution and image sequence length. Our



(a) Filtered feature maps of the first convolutional layer



(b) Sky image filtered feature maps in deep learning hierarchical representations (C1–C13)

Fig. 13. Filtered sky image feature maps by the 10 MA SCNN model. A sky image at 2016-06-19 05:00:00 is randomly selected for the demonstration. All the 64 filtered feature maps are included in (a) to show the diversity of filters. A feature map is selected from each convolutional layer to show the feature development by hierarchical representations in (b), where feature maps are grouped in columns by the feature learning block and in rows by the layer order.

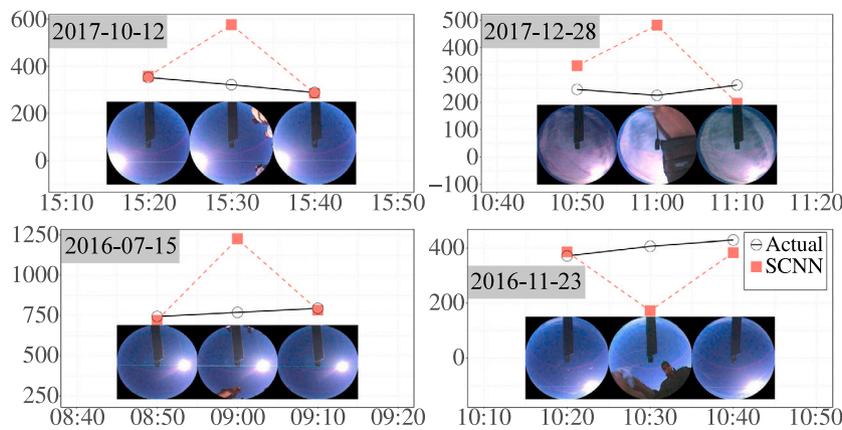


Fig. 14. Defective sky images.

research found that using two images with a resolution of 128×128 provided the most accurate forecasts at the case study location. We also interpreted the feature attributions that CNN models learned by the feature map visualization.

CRedit authorship contribution statement

Cong Feng: Conceptualization, Methodology, Software, Validation, Writing – original draft, Visualization. **Jie Zhang:** Conceptualization, Resources, Writing – review & editing, Supervision, Project administration. **Wenqi Zhang:** Conceptualization, Investigation, Writing – review & editing, Visualization. **Bri-Mathias Hodge:** Resources, Writing – review & editing, Supervision, Project administration.

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.

Acknowledgments

This work was authored in part by the National Renewable Energy Laboratory (NREL), USA, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract NO. DE-AC36-08GO28308. Funding provided through a NREL’s LDRD program called the Director’s Postdoctoral Fellowships. The views expressed

in the article do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes.

The authors thank Dr. Christopher S. Simmons from UT Dallas for helping set up the two HPC systems. The authors would also like to acknowledge the TACC at UT System for providing HPC resources that have contributed to the research results reported within this paper.

References

- [1] Inman Rich H, Pedro Hugo TC, Coimbra Carlos FM. Solar forecasting methods for renewable energy integration. *Prog Energy Combust Sci* 2013;39(6):535–76.
- [2] Yang Dazhi, Kleissl Jan, Gueymard Christian A, Pedro Hugo TC, Coimbra Carlos FM. History and trends in solar irradiance and PV power forecasting: A preliminary assessment and review using text mining. *Sol Energy* 2018;168:60–101.
- [3] Diagne Maimouna, David Mathieu, Lauret Philippe, Boland John, Schmutz Nicolas. Review of solar irradiance forecasting methods and a proposition for small-scale insular grids. *Renew Sustain Energy Rev* 2013;27:65–76.
- [4] Antonanzas Javier, Osorio Natalia, Escobar Rodrigo, Urraca Ruben, Martinez-de-Pison Francisco J, Antonanzas-Torres Fernando. Review of photovoltaic power forecasting. *Sol Energy* 2016;136:78–111.
- [5] Aguiar L Mazorra, Pereira Brais, Lauret Philippe, Díaz F, David Mathieu. Combining solar irradiance measurements, satellite-derived data and a numerical weather prediction model to improve intra-day solar forecasting. *Renew Energy* 2016;97:599–610.
- [6] Wolff Björn, Kühnert Jan, Lorenz Elke, Kramer Oliver, Heinemann Detlev. Comparing support vector regression for PV power forecasting to a physical modeling approach using measurement, numerical weather prediction, and cloud motion data. *Sol Energy* 2016;135:197–208.
- [7] Ferlito S, Adinolfi G, Graditi G. Comparative analysis of data-driven methods online and offline trained to the forecasting of grid-connected photovoltaic plant production. *Appl Energy* 2017;205:116–29.
- [8] Elsinga Boudewijn, van Sark Wilfried GJHM. Short-term peer-to-peer solar forecasting in a network of photovoltaic systems. *Appl Energy* 2017;206:1464–83.
- [9] Voyant Cyril, Notton Gilles. Solar irradiation nowcasting by stochastic persistence: A new parsimonious, simple and efficient forecasting tool. *Renew Sustain Energy Rev* 2018;92:343–52.
- [10] ISO/RTO council white paper: Variable energy resources, system operations and wholesale markets. Technical report, The IRC Markets Committee; 2010.
- [11] Glassley William, Kleissl Jan, Shiu Henry, Huang Junhui. California renewable energy forecasting, resource data and mapping. Technical report, Regents of the University of California; 2012.
- [12] Pothineni Dinesh, Oswald Martin R, Poland Jan, Pollefeys Marc. Kloudnet: Deep learning for sky image analysis and irradiance forecasting. In: German conference on pattern recognition. Springer; 2018, p. 535–51.
- [13] Zhao Xin, Wei Haikun, Wang Hai, Zhu Tingting, Zhang Kanjian. 3D-CNN-based feature extraction of ground-based cloud images for direct normal irradiance prediction. *Sol Energy* 2019;181:510–8.
- [14] Sun Yuchi, Szűcs Gergely, Brandt Adam R. Solar PV output prediction from video streams using convolutional neural networks. *Energy Environ Sci* 2018;11(7):1811–8.
- [15] Sun Yuchi, Venugopal Vignesh, Brandt Adam R. Short-term solar power forecast with deep learning: Exploring optimal input and output configuration. *Sol Energy* 2019;188:730–41.
- [16] Feng Cong, Cui Mingjian, Hodge Bri-Mathias, Lu Siyuan, Hamann Hendrik F, Zhang Jie. Unsupervised clustering-based short-term solar forecasting. *IEEE Trans Sustain Energy* 2018;10(4):2174–85.
- [17] Chu Yinghao, Pedro Hugo TC, Coimbra Carlos FM. Hybrid intra-hour DNI forecasts with sky image processing enhanced by stochastic learning. *Sol Energy* 2013;98:592–603.
- [18] Marquez Ricardo, Coimbra Carlos FM. Intra-hour DNI forecasting based on cloud tracking image analysis. *Sol Energy* 2013;91:327–36.
- [19] Li Mengying, Chu Yinghao, Pedro Hugo TC, Coimbra Carlos FM. Quantitative evaluation of the impact of cloud transmittance and cloud velocity on the accuracy of short-term DNI forecasts. *Renew Energy* 2016;86:1362–71.
- [20] Antipov Grigory, Berrani Sid-Ahmed, Ruchaud Natacha, Dugelay Jean-Luc. Learned vs. hand-crafted features for pedestrian gender recognition. In: Proceedings of the 23rd ACM international conference on multimedia, 2015, p. 1263–6.
- [21] Cruciani Federico, Vafeiadis Anastasios, Nugent Chris, Cleland Ian, McCullagh Paul, Votis Konstantinos, et al. Comparing CNN and human crafted features for human activity recognition. In: 2019 IEEE smartworld, ubiquitous intelligence & computing, advanced & trusted computing, scalable computing & communications, cloud & big data computing, internet of people and smart city innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI). IEEE; 2019, p. 960–7.
- [22] Hssayeni Murtadha D, Saxena Sagar, Ptucha Raymond, Savakis Andreas. Distracted driver detection: Deep learning vs handcrafted features. *Electron Imaging* 2017;2017(10):20–6.
- [23] Collobert Ronan, Weston Jason, Bottou Léon, Karlen Michael, Kavukcuoglu Koray, Kuksa Pavel. Natural language processing (almost) from scratch. *J Mach Learn Res* 2011;12(ARTICLE):2493–537.
- [24] Krizhevsky Alex, Sutskever Ilya, Hinton Geoffrey E. Imagenet classification with deep convolutional neural networks. *Adv Neural Inf Process Syst* 2012;25:1097–105.
- [25] Mnih Volodymyr, Kavukcuoglu Koray, Silver David, Rusu Andrei A, Veness Joel, Bellemare Marc G, et al. Human-level control through deep reinforcement learning. *Nature* 2015;518(7540):529–33.
- [26] Silver David, Huang Aja, Maddison Chris J, Guez Arthur, Sifre Laurent, Van Den Driessche George, et al. Mastering the game of go with deep neural networks and tree search. *Nature* 2016;529(7587):484–9.
- [27] Kong Weicong, Jia Youwei, Dong Zhao Yang, Meng Ke, Chai Songjian. Hybrid approaches based on deep whole-sky-image learning to photovoltaic generation forecasting. *Appl Energy* 2020;280:115875.
- [28] Zhen Zhao, Liu Jiaming, Zhang Zhanyao, Wang Fei, Chai Hua, Yu Yili, et al. Deep learning based surface irradiance mapping model for solar PV power forecasting using sky image. *IEEE Transactions On Industry Applications* 2020;56(4):3385–96.
- [29] Lathuilière Stéphane, Mesejo Pablo, Alameda-Pineda Xavier, Horaud Radu. A comprehensive analysis of deep regression. *IEEE Trans Pattern Anal Mach Intell* 2019;42(9):2065–81.
- [30] Zhang Ruiyuan, Ma Hui, Saha Tapan K, Zhou Xiaofang. Photovoltaic nowcasting with bi-level spatio-temporal analysis incorporating sky images. *IEEE Trans Sustain Energy* 2021.
- [31] Paletta Quentin, Lasenby Joan. Convolutional neural networks applied to sky images for short-term solar irradiance forecasting. 2020, arXiv preprint arXiv:2005.11246.
- [32] Ryu Anto, Ito Masakazu, Ishii Hideo, Hayashi Yasuhiro. Preliminary analysis of short-term solar irradiance forecasting by using total-sky imager and convolutional neural network. In: 2019 IEEE PES GTD grand international conference and exposition Asia (GTD Asia). IEEE; 2019, p. 627–31.
- [33] Wang Fei, Xuan Zhiming, Zhen Zhao, Li Yu, Li Kangping, Zhao Liqiang, et al. A minutely solar irradiance forecasting method based on real-time sky image-irradiance mapping model. *Energy Convers Manage* 2020;220:113075.
- [34] Yang Dazhi. A guideline to solar forecasting research practice: Reproducible, operational, probabilistic or physically-based, ensemble, and skill (ROPES). *J Renew Sustain Energy* 2019;11(2):022701.
- [35] Feng Cong, Zhang Jie. Solarnet: A sky image-based deep convolutional neural network for intra-hour solar forecasting. *Sol Energy* 2020;204:71–8.
- [36] Ozcanli Asiyeh K, Yaprakdal Fatma, Baysal Mustafa. Deep learning methods and applications for electrical power systems: A comprehensive review. *Int J Energy Res* 2020;44(9):7136–57.
- [37] Mishra Manohar, Nayak Janmenjoy, Naik Bighnaraj, Abraham Ajith. Deep learning in electrical utility industry: A comprehensive review of a decade of research. *Eng Appl Artif Intell* 2020;96:104000.
- [38] Khodayar Mahdi, Liu Guangyi, Wang Jianhui, Khodayar Mohammad E. Deep learning in power systems research: A review. *CSEE J Power Energy Syst* 2020.
- [39] Zhang Dongxia, Han Xiaoqing, Deng Chunyu. Review on the research and practice of deep learning and reinforcement learning in smart grids. *CSEE J Power Energy Syst* 2018;4(3):362–70.
- [40] Al-Saffar Ahmed Ali Mohammed, Tao Hai, Talab Mohammed Ahmed. Review of deep convolution neural network in image classification. In: 2017 International conference on radar, antenna, microwave, electronics, and telecommunications. IEEE; 2017, p. 26–31.
- [41] Rawat Waseem, Wang Zenghui. Deep convolutional neural networks for image classification: A comprehensive review. *Neural Comput* 2017;29(9):2352–449.
- [42] Simonyan Karen, Zisserman Andrew. Very deep convolutional networks for large-scale image recognition. 2014, arXiv preprint arXiv:1409.1556.
- [43] He Kaiming, Zhang Xiangyu, Ren Shaoqing, Sun Jian. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, p. 770–8.
- [44] Cai Shaofeng, Shu Yao, Chen Gang, Ooi Beng Chin, Wang Wei, Zhang Meihui. Effective and efficient dropout for deep convolutional neural networks. 2019, arXiv preprint arXiv:1904.03392.
- [45] Stoffel T, Andreas A. Nrel solar radiation research laboratory (srri): Baseline measurement system (bms); golden, colorado (data). Technical report, National Renewable Energy Lab.(NREL), Golden, CO (United States); 1981.
- [46] Feng Cong, Yang Dazhi, Hodge Bri-Mathias, Zhang Jie. Opensolar: Promoting the openness and accessibility of diverse public solar datasets. *Sol Energy* 2019;188:1369–79.

- [47] Stein Joshua S, Holmgren William F, Forbess Jessica, Hansen Clifford W. PVLIB: Open source photovoltaic performance modeling functions for matlab and python. In: 2016 IEEE 43rd photovoltaic specialists conference. IEEE; 2016, p. 3425–30.
- [48] Bergmeir Christoph Norbert, Benítez Sánchez José Manuel, et al. Neural networks in R using the Stuttgart neural network simulator: RSNNS, Vol. 46. (7). American Statistical Association; 2012.
- [49] Ripley Brian, Venables William. Nnet: Feed-forward neural networks and multinomial log-linear models. R Package Version 2016;7.
- [50] Ridgeway Greg. Generalized boosted models: A guide to the gbm package. Update 2007;1(1):2007.
- [51] Kuhn Max. Caret: classification and regression training. Astrophys Source Code Libr 2015;ascl-1505.
- [52] Liaw Andy, Wiener Matthew, et al. Classification and regression by randomforest. R News 2002;2(3):18–22.
- [53] Feng Cong, Cui Mingjian, Lee Meredith, Zhang Jie, Hodge Bri-Mathias, Lu Siyuan, et al. Short-term global horizontal irradiance forecasting based on sky imaging and pattern recognition. In: 2017 IEEE power & energy society general meeting. IEEE; 2017, p. 1–5.
- [54] Feng Cong, Zhang Jie. Hourly-similarity based solar forecasting using multi-model machine learning blending. In: 2018 IEEE power & energy society general meeting. IEEE; 2018, p. 1–5.
- [55] Yang Dazhi, Alessandrini Stefano, Antonanzas Javier, Antonanzas-Torres Fernando, Badescu Viorel, Beyer Hans Georg, et al. Verification of deterministic solar forecasts. Sol Energy 2020;210:20–37.
- [56] Shankar Vaishaal, Roelofs Rebecca, Mania Horia, Fang Alex, Recht Benjamin, Schmidt Ludwig. Evaluating machine accuracy on imagenet. In: International conference on machine learning. PMLR; 2020, p. 8634–44.
- [57] Sun Mucun, Feng Cong, Zhang Jie. Probabilistic solar power forecasting based on weather scenario generation. Appl Energy 2020;266:114823.
- [58] Dev Soumyabrata, Savoy Florian M, Lee Yee Hui, Winkler Stefan. WAHRSIS: A low-cost high-resolution whole sky imager with near-infrared capabilities. In: Infrared imaging systems: Design, analysis, modeling, and testing XXV, Vol. 9071. International Society for Optics and Photonics; 2014, p. 90711L.
- [59] Mammoli Andrea, Ellis Abraham, Menicucci Anthony, Willard Steve, Caudell Thomas, Simmins John. Low-cost solar micro-forecasts for PV smoothing. In: 2013 1st IEEE conference on technologies for sustainability. IEEE; 2013, p. 238–43.
- [60] Sun Xixi, Bright Jamie M, Gueymard Christian A, Acord Brendan, Wang Peng, Engerer Nicholas A. Worldwide performance assessment of 75 global clear-sky irradiance models using principal component analysis. Renew Sustain Energy Rev 2019;111:550–70.
- [61] Dev Soumyabrata, Savoy Florian M, Lee Yee Hui, Winkler Stefan. Design of low-cost, compact and weather-proof whole sky imagers for high-dynamic-range captures. In: 2015 IEEE international geoscience and remote sensing symposium. IEEE; 2015, p. 5359–62.