Short-Term Global Horizontal Irradiance Forecasting Using a Hybrid Convolutional Neural Network-Gate Recurrent Unit Method

Yu Zhang, Jianfeng Ma, Cong Zeng and Guangyao Li

College of Electronics and Information Engineering, Tongji University, Shanghai
201804, China
Email: lgy423@126.com

Abstract. The effectiveness of photovoltaic power generation systems—a clean and renewable use of solar energy—depends on the amount of solar radiation. Consequently, solar radiation forecasting (especially short-term) is crucial for photovoltaic plants. In this paper, a hybrid convolutional neural network (CNN) and gate recurrent unit (GRU) method is proposed for short-term (10-min) solar radiation forecasting based on image and time-series data (i.e., radiation data at different times). The method aims to achieve high performance in solar radiation forecasting, which can be useful for PV plant adjustment. CNN–GRU consists of two branches. One is based on the ResNet-18 structure, which can extract features from sky images. The other is a GRU branch, which consists of three fully connected layers used for meteorological feature extraction. Experiments on a public dataset showed that our method predicts the mean absolute error better than other benchmark models. The ablation experiments demonstrated that the hybrid model performs better than a single model and, therefore, shows promise for application in solar radiation forecasting.

Keywords. Solar forecasting; global horizontal irradiance; deep learning; hybrid model.

1. Introduction

Energy supply is indispensable for societal and economic activities. Traditional energy resources, such as oil, coal, and gas, face the risks of depletion and environmental pollution through harmful emissions. Consequently, scientists and researchers have endeavoured to find renewable and sustainable energy sources. One of the most promising, solar energy, has the advantages of being abundant, free, and non-polluting. Thus, solar photovoltaic (PV) power generation systems have received increasing attention, and PV generation has grown exponentially in the last decade. However, PV generation depends on solar radiation and is affected by its fluctuations. Therefore, plant operators need PV plant output forecasts to bid in energy markets, and accurate solar radiation forecasting is crucial for PV plant output. Hence, researchers are seeking a low-cost and efficient method for radiation forecasting.

Some early methods require manual or complicated operations, such as sun area determination [1], cloudiness type classification [2], or vector division [3], which limit their practical application.

Zapata et al. [4] used optical flow for cloud tracking and short-term forecasting. Richardson et al. [5] collected all-sky images (ASIs) for cloud tracking and used optical flow for intra-hour irradiance forecasting.

In recent years, machine learning, especially deep learning, has made great strides in a number of fields, including computer vision and natural language processing. Deep learning has attracted
attention owing to its great nonlinear ability, and researchers have increasingly focused on deep learning-based radiation forecasting.

Kamadinata et al. [6] used artificial neural network models to manage image and irradiance data for radiation forecasting.

Many different models for estimating solar radiation at different time intervals (monthly, daily, and hourly) were reviewed by Zhang et al. [7]. Detailed machine learning reviews have been reported in recent papers [8, 9].

Numerous convolutional neural network (CNN)-based methods have been proposed for solar radiation forecasting. For example, SolarNet [10] was developed to forecast the operational “1-h-ahead” global horizontal irradiance using only sky images without numerical measurements and extra feature engineering, which proved superior under various weather conditions. Leelaruji and Teerakawanich [11] used a CNN-based on ResNet [12] to raise trigger events 1–2 min before sun cover occurs.

As most of the above-mentioned deep learning methods only consider one type of data (e.g., image, time-series, or structured weather), the full simultaneous use of image and time-series data cannot be achieved. Deep learning studies have used hybrid models that utilize image and time-series data, such as those proposed by Zang et al. [13] and Kong et al. [14]. Our study is inspired by the work of Zang et al. [13], who applied the CNN-long short-term memory (LSTM) method to predict solar irradiance, and the work of Kong et al. 20, who proposed a hybrid version of the static image only (SIO)—the sky image hybrid (SIH) model—for photovoltaic generation forecasting. The SIO consists of three hidden convolutional layers and two hidden fully connected layers, which may not be sufficient to extract the features of the image. Thus, we propose a hybrid CNN–GRU model that makes full simultaneous use of image and time-series data. The CNN–GRU model has fewer parameters and is faster than the CNN–LSTM model. We selected ResNet-18 as the CNN module because of its super-feature extraction and nonlinear mapping abilities. As the original ResNet-18 is used for 1,000 categories of object classification, we changed some layers of ResNet-18 and adjusted the model to suit our radiation forecasting requirements. In addition, we applied the GRU for time-series data feature extraction. Figure 1 shows the detailed model structure. The experiments performed on the public dataset demonstrated that the proposed model had a promising prediction performance. Our contributions include the following:

- We formulated solar irradiance forecasting as the regression task and used the hybrid CNN–GRU method to complete the task, by integrating image and time-series data for better performance.
- To confirm performance, we compared our proposed model with different benchmark models, including the persistence, moving average (MA) and machine learning models. We also performed ablation experiments to compare the hybrid and single models. The results showed that our model exhibited better performance. The model was also tested under different weather conditions.

2. Methods

In this section, we first introduce the framework of our research and then formulate the ASI-based 10-min solar forecasting problem. Subsequently, we introduce the entire framework of the proposed model. Finally, we explain the prediction method for solar radiation forecasting.

2.1. Research Framework

Figure 2 shows a schematic of our research, which consists of three parts: data, model, and application; this paper primarily focuses on the second part. The image and time-series data were from a public dataset collected by the sensors. Details of the dataset are introduced in Section III. Our model uses the obtained data to improve with constant training and outputs solar radiation predictions; the results can guide the adjustment of PV plants and lead to stable power generation.
2.2. Problem Formulation
Solar radiation is affected by many factors; cloudiness is the most crucial, as it decreases radiation. We evaluate the level of radiation using sky images. In addition, CNN extracts features from an image and maps the nonlinear function. Because the radiation value at time $t$ can be influenced by the radiation value before time $t$ (i.e., at time $t-1$, $t-2$... $t-n$), time-series data is also important; therefore, we applied the GRU for numerical time-series data feature extraction. The sky images and time-series data for determining radiation are mapped with the use of CNN-GRU using the following formulas:

$$P = f(X)$$
$$Q = g(D)$$
$$O = h(\text{concat}(P, Q))$$

where $X \in \mathbb{R}^{N \times C \times H \times W}$ denotes the image batch; $N$, $C$, $H$, and $W$ are the batch size, image channel, image height, and image width, respectively. $P \in \mathbb{R}^{N \times 1}$, where $P$ denotes the corresponding mediate output; $f$ denotes the mapping of the CNN; $D \in \mathbb{R}^{w \times 1}$, where $w$ is the window size of the time-series data; $Q \in \mathbb{R}^{N \times 1}$, where $Q$ denotes the corresponding mediate output of the GRU; and $g$ denotes the GRU mapping. The final output is the full connection of the combination of the two above-mentioned mediated outputs. Thus, we applied the CNN and GRU to map the image and time-series data to predict the radiation. The final loss function is given by equation (4):

$$\text{loss} = \sum_{i=0}^{n} (o_i - \hat{o}_i)^2 + \frac{\lambda}{2n} \|w\|^2$$

where $o_i$ denotes the $i$-th output of the model, $\hat{o}_i$ denotes the corresponding $i$-th ground truth, $W$ denotes the weight matrix, and $n$ denotes the total number of data. $\lambda$ is a hyperparameter that balances the importance of error and weight.

2.3. Deep Learning Model
Figure 2 shows the model structure. The entire framework can be divided into the GRU and CNN branches. The input of the GRU branch is time-series data consisting of two GRU layers. The input of the GRU branch is a five-dimensional tensor (time-series data), whereas the output of the GRU branch is a one-dimensional tensor. The CNN branch obtains image data as an input. In this branch, we first preprocess the ASIs and then feed them into the ResNet-18 layer, which extracts the image features. As the original ResNet-18 is used for the image classification task and outputs a thousand-dimensional tensor, we used the partial structure of ResNet-18 and discarded the final fully connected layers of ResNet-18. Then, we flattened the 512-dimensional ResNet-18 and added a full-connect layer (FC-1), which indicates that the output dimension of the layer is 1. Thus, the output of the CNN branch is a one-dimensional tensor. Subsequently, we combined the outputs of the GRU and CNN into a 2-
dimensional tensor. Finally, we used an FC-1 to map a two-dimensional tensor into the final result. The FC-1 implied that the output dimension of this layer was 1.

We set the layer number of the GRU to two. The use of multiple layers makes GRU training more difficult. As the number of layers increases, gradient disappearance or gradient explosion are more likely to occur. Consequently, the loss does not decrease or converge.

For the GRU branch, each layer computes the following function for each element in the input sequence (Figure 3 illustrates the internal structure of the GRU):

\[
\begin{align*}
    r_t &= \sigma(W_{ir}x_t + b_{ir} + W_{hr}h_{t-1} + b_{hr}) \\
    z_t &= \sigma(W_{iz}x_t + b_{iz} + W_{hz}h_{t-1} + b_{hz}) \\
    n_t &= \tanh(W_{in}x_t + b_{in} + r_t \cdot (W_{hn}h_{t-1} + b_{hn})) \\
    h_t &= (1 - z_t) \cdot n_t + z_t \cdot h_{t-1}
\end{align*}
\]

where \(h_t\) is the hidden state at time \(t\); \(x_t\) is the input at time \(t\); \(h_{t-1}\) is the hidden state of the layer at time \(t\) or the initial hidden state at time \(0\); and \(r_t\), \(z_t\), and \(n_t\) are the reset, update, and new gates, respectively. \(\sigma\) is the sigmoid function, and \(\ast\) is the Hadamard product. \(W\) is a parameter matrix that can be trained by the model.

![Figure 2. Schematic diagram for our research. From top to bottom are the data (red dotted box), model (blue dotted box), and application (green dotted box) parts [best viewed in color].](image)

![Figure 3. Internal structure of the GRU.](image)

3. Experiments
In this section, we briefly introduce the dataset and software used, the parameters set during the training process, and the benchmark models used for comparison.
3.1. Datasets and Software
We used the public cloud image National Renewable Energy Laboratory (NREL)—Solar Radiation Research Laboratory (SRRL) dataset [15]. The NREL-SRRL has been collecting continuous solar measurements at the South Table Mountain Campus of NREL (longitude: 105.18° W, latitude: 39.74° N, and elevation: 1,828.2 m) in Golden, Colorado, USA since 1981 [15]. The size of the images was 1,536 pixels × 1,536 pixels, and the images were captured by an ASI-16 camera beginning September 26, 2017. The paper by Feng et al. [15] contains a detailed description of and an introduction to the dataset. The image data can be downloaded automatically in batches with Python Spyder using third-party libraries, such as urllib2, request, and BeautifulSoup. The data can be easily downloaded by selecting the start and end dates.

3.2. Benchmark Models
To determine the performance, we compared our model with benchmark models, including the MA and persistence models [16]. The MA model averages the historical data and obtains the results according to the size of the sliding window. The persistence model is a widely used benchmark model for short-term solar irradiation forecasting, which assumes that the irradiation data at time $t$ are the same as those at $t−1$. Here, we assume that the radiation data at time $t$ are the same as those at time $t−2$.

In addition to the models mentioned above, we also use some machine learning models as the benchmark models, such as K-Nearest Neighbor (KNN), Support Vector Machine Regression (SVR) and Gradient Boosting Decision Tree (GBDT).

4. Results
In this section, we first describe the evaluation matrix used to evaluate the performance of the different models. Then, the results of three different types of experiments (including parameter optimization, comparison, and ablation) are discussed. Subsequently, an illustration of different model performance comparisons is presented. Finally, the results of the model performance are obtained, and conclusions are derived.

4.1. Evaluation Matrix
To evaluate the model performance, we used three different evaluation methods: mean absolute error (MAE), root mean squared error (RMSE), and correlation (CORR).

4.2. Comparison
We downloaded the images from January 1, 2020, to November 18, 2020. For the downloaded data, at some time $t$, there were a total of six different images: (a) Original normal exposed image, (b) calibrated normal exposed image, (c) original underexposed image, (d) calibrated underexposed image, (e) BRBG-processed image, and (f) CDOC-processed image.

In our research, the original normal exposed image was taken as our image data for normal exposure and maximum reservation of the original information. Consequently, the other images (e.g., pre-processed, underexposed, and calibrated) may lose some important information and harm the model performance. The image taken at time $t$ matches the radiation data at time $t$ in the radiation data file. Thus, we observed one of every six images; some images were not useful in our research. Thus, the total number of images was 22,821.

To assess the performance robustness of the models in the test dataset, we selected 1/7 of all images for the test dataset to cover all seasons and weather conditions, including sunny, cloudy, and less cloudy. We split the images into training, validation, and test sets with 17,605, 1956, and 3,260 images, respectively.

The training set was used to train the model, whereas the validation set was used to change and save the model parameters. A test dataset was used to evaluate the generalization of the model.

To confirm the performance of the CNN–GRU model for solar irradiance, we compared the different models using the different time interval datasets, including 10mins, 20 mins, 30 mins, 40 mins, 50 mins and 60 mins. Table 1 shows the results.
### Table 1. Prediction performance comparison of different models in different time intervals datasets.

| Time interval | Method                        | MAE     | RMSE     | CORR     |
|---------------|-------------------------------|---------|----------|----------|
|               | Persistence model (2)         | 56.89219| 84.43951 | 0.93205  |
|               | Persistence model (3)         | 74.81597| 101.60805| 0.90167  |
| 10-min        | MA (window = 4)               | 60.52395| 81.01723 | 0.93627  |
|               | MA (window = 5)               | 68.58341| 88.54946 | 0.92356  |
|               | SVR                           | 61.09137| 123.52374| 0.91591  |
|               | GBRT                          | 66.09336| 123.23370| 0.91653  |
|               | KNN                           | 62.41719| 122.77719| 0.91724  |
|               | CNN–GRU                       | **51.29294**| **70.49027**| **0.96146** |
|               | Persistence model (2)         | 96.59655| 130.70770| 0.855787 |
|               | Persistence model (3)         | 128.19425| 160.93127| 0.781518 |
| 20-min        | MA (window = 4)               | 104.12128| **129.60091**| 0.852398 |
|               | MA (window = 5)               | 117.67788| 143.62527| 0.816659 |
|               | SVR                           | 86.52597| 152.31378| 0.869415 |
|               | GBRT                          | 89.57771| 149.50084| 0.873944 |
|               | KNN                           | 83.52016| 149.28916| 0.875240 |
|               | CNN–GRU                       | **78.99216**| 133.13096| **0.889000** |
|               | Persistence model (2)         | 128.12624| 161.13057| 0.78268  |
| 30-min        | Persistence model (3)         | 171.18689| 206.64107| 0.64290  |
|               | MA (window = 4)               | 138.35508| 166.57417| 0.75286  |
|               | MA (window = 5)               | 156.90723| 186.89816| 0.68208  |
|               | SVR                           | 93.68878| 158.93349| 0.86109  |
|               | GBRT                          | 98.36217| 157.89106| 0.86229  |
|               | KNN                           | 93.70198| 155.97630| 0.86617  |
|               | CNN–GRU                       | **76.99806**| 125.51898| **0.87650** |
|               | Persistence model (2)         | 159.00588| 194.56287| 0.68817  |
| 40-min        | Persistence model (3)         | 212.40374| 252.75245| 0.47438  |
|               | MA (window = 4)               | 171.75506| 204.63134| 0.62139  |
|               | MA (window = 5)               | 193.62525| 228.58302| 0.51023  |
|               | SVR                           | 99.88928| 159.28548| 0.85858  |
|               | GBRT                          | 109.76753| 164.32165| 0.84843  |
|               | KNN                           | 105.28971| 159.49655| 0.85923  |
|               | CNN–GRU                       | **59.71306**| **93.91760**| **0.93005** |
|               | Persistence model (2)         | 180.00254| 214.90182| 0.611031 |
| 50-min        | Persistence model (3)         | 245.0308| 290.15755| 0.292515 |
|               | MA (window = 4)               | 194.93950| 231.16091| 0.493350 |
|               | MA (window = 5)               | 216.51907| 255.30803| 0.34928  |
|               | SVR                           | 102.58932| 167.10860| 0.84220  |
|               | GBRT                          | 114.94254| 169.89452| 0.84089  |
|               | KNN                           | 105.28971| 167.16739| 0.84625  |
|               | CNN–GRU                       | **50.52559**| **76.97907**| **0.93785** |
|               | Persistence model (2)         | 211.15186| 252.80381| 0.48482  |
| 60-min        | Persistence model (3)         | 281.24166| 332.96039| 0.10852  |
|               | MA (window = 4)               | 222.03769| 262.89693| 0.34462  |
|               | MA (window = 5)               | 242.16955| 284.66351| 0.17651  |
|               | SVR                           | 103.84116| 165.37649| 0.84929  |
|               | GBRT                          | 110.81171| 167.33639| 0.83741  |
|               | KNN                           | 105.71269| 166.91645| 0.83941  |
|               | CNN–GRU                       | **53.13291**| **82.48329**| **0.92560** |
As other models get worse when the time interval increase, CNN-GRU model keeps its performance. Besides, it always gets the best performance compared with other benchmark models.

4.3. Ablation Study
Ablation studies are used to describe the process of removing certain parts of the network to better understand network behaviour. The ablation experiment was designed to illustrate the role and contribution of each part of the model. Specifically, if the network performance deteriorates sharply when module A in the network is removed, module A contributes significantly to the performance of the network. Therefore, in this study, we investigated the effect of the CNN and GRU branches to better understand the effectiveness of the CNN-GRU model. To demonstrate the importance of each part of the CNN–GRU, we conducted ablation experiments. We compared the performances of the CNN–GRU, CNN, and GRU models using the test dataset. The function of the ablation experiment was to confirm that the performance of the CNN-GRU model was better than that of the single CNN or GRU models. Table 2 shows the results.

| Method   | MAE    | RMSE  | CORR  |
|----------|--------|-------|-------|
| CNN      | 87.79172 | 126.93450 | 0.89054 |
| GRU      | 71.92086 | 90.86387  | 0.93072 |
| CNN-GRU  | 51.29294 | 70.49027  | 0.96146 |

The MAE, RMSE, and CORR performances of the hybrid CNN–GRU were better than that of the other two single models, because the hybrid model used image and time-series data. The hybrid model improved the MAE and RMSE by at least 28% and 22%, respectively, compared to the single models.

4.4. Prediction Visualization
To prove the efficiency of the CNN-GRU model, we illustrated its prediction performance. Figure 4 shows the results.

![Figure 4](image)

Figure 4. Prediction results of the ablation experiment. The blue line is the ground truth, and the red line is the prediction of the CNN-GRU model [best viewed in color].

The illustration reveals that the hybrid model almost coincides with the ground truth, because the CNN-GRU model uses both image and time-series data, thereby achieving better performance results.

5. Conclusions and Future Work
In this study, we formulated a prediction as a regression task. We also proposed a hybrid CNN–GRU method for irradiance forecasting, which achieved good performance on the NREL public online dataset and showed promising application value. Our study can provide a reference and baseline for
other researchers, especially for those who want to evaluate the performance of the models proposed on the same dataset.

Our future research will improve this study by: (1) incorporating more spatial, temporal, or weather data; (2) using other advanced methods, such as the attention mechanism or image pyramid; and (3) obtaining more data to promote the generalization of the model and to apply our model to new datasets.

Acknowledgments
This work was supported by the National Natural Science Foundation of China (NSFC) (grant number 61771346). We would also like to thank Qiaochuan Chen for proofreading the article.

References
[1] Alonso J, Batlles F J, Villarroel C, Ayala R and Burgaleta J I 2014 Determination of the sun area in sky camera images using radiometric data Energy Convers and Manag. 78 24-31.
[2] Alonso-Montesinos J and Batlles F J 2015 Solar radiation forecasting in the short- and medium-term under all sky conditions Energy 83 387-393.
[3] Alonso J and Batlles F J 2014 Short and medium-term cloudiness forecasting using remote sensing techniques and sky camera imagery Energy 73 890-897.
[4] Wood-Bradley P, Zapata J and Pye J 2012 Cloud tracking with optical flow for short-term solar forecasting 50th Conf. Aust. Sol. Energy Soc. pp 2-7.
[5] Richardson W, Krishnaswami H, Vega R and Cervantes M 2017 A low cost, edge computing, all-sky imager for cloud tracking and intra-hour irradiance forecasting Sustain. 9 (4) 1-17.
[6] Kamadinata J O, Ken T L and Suwa T 2019 Sky image-based solar irradiance prediction methodologies using artificial neural networks Renew. Energy 134 837-845.
[7] Zhang J, Zhao L, Deng S, Xu W and Zhang Y 2017 A critical review of the models used to estimate solar radiation Renew. Sustain. Energy Rev. 70 314-329.
[8] Cyril V, Gilles N, Soteris K, Marie-Laure N, Christophe P, Fabrice M and Alexis F 2017 Machine Learning methods for solar radiation forecasting: A review Renew. Energy 105 569-582.
[9] Yang D, Kleissl J, Gueymard C A, Pedro H T C and Coimbra C F M 2018 History and trends in solar irradiance and PV power forecasting: A preliminary assessment and review using text mining Sol. Energy 168 60-101.
[10] Feng C and Zhang J 2020 SolarNet: A sky image-based deep convolutional neural network for intra-hour solar forecasting Sol. Energy 204 71-78.
[11] Leelaruji T and Teerakawanich N 2020 Short term prediction of solar irradiance fluctuation using image processing with ResNet 2020 8th Int. Electr. Eng. Congr. iECON pp 1-4.
[12] He K, Zhang X, Ren S and Sun J 2016 Deep residual learning for image recognition Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. pp 770-778.
[13] Zang H, Liu L, Sun L, Cheng L, Wei Z and Sun G 2020 Short-term global horizontal irradiance forecasting based on a hybrid CNN-LSTM model with spatiotemporal correlations Renew. Energy 160 26-41.
[14] Kong W, Jia Y, Dong Z Y, Meng K and Chai S 2020 Hybrid approaches based on deep whole-sky-image learning to photovoltaic generation forecasting Appl. Energy 280 115875.
[15] Feng C, Yang D, Hodge B M and Zhang J 2019 OpenSolar: Promoting the openness and accessibility of diverse public solar datasets Sol. Energy 188 1369-1379.
[16] Almeida M P, Perpiñán O and Narvarte L 2015 PV power forecast using a nonparametric PV model Sol. Energy 115 354-368.