Solar Energy 267 (2024) 112203

Contents lists available at ScienceDirect

Solar Energy

journal homepage: www.elsevier.com/locate/solener

On vision transformer for ultra-short-term forecasting of photovoltaic generation using sky images

Shijie Xu [a], Ruiyuan Zhang [b], Hui Ma [a,*], Chandima Ekanayake [a], Yi Cui [c]

[a] School of Information Technology and Electrical Engineering, University of Queensland, Brisbane, 4072, Australia
[b] The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong, China
[c] University of Southern Queensland, Brisbane, Australia

ARTICLE INFO

Keywords:
Deep learning
Forecasting
Image processing
Photovoltaic
Vision transformers

ABSTRACT

An accurate photovoltaic (PV) generation forecasting is important for grid scheduling and dispatching. However, ultra-short-term PV generation forecasting is rather challenging because weather conditions may change significantly in a short time period largely due to the dynamics and movement of clouds above a solar PV farm. For monitoring clouds above the solar PV farm, ground-based whole-sky cameras (Sky-Imagers) have been installed. This paper develops a novel cloud image-based ultra-short-term forecasting framework. Within the framework, an integration of the Vision Transformer (ViT) model and the Gated Recurrent Unit (GRU) encoder is designed for the high-dimensional latent feature analysis. A Multi-Layer Perception (MLP) is employed to generate the one-step-ahead PV generation forecasting. Numeric experiments are conducted using real-world solar PV datasets. The results show that the proposed framework and algorithms can achieve higher accuracy compared to several baseline methods for ultra-short-term PV generation forecasting.

1. Introduction

With the growing integration of solar photovoltaic (PV) plants into electricity networks, it is necessary to provide an accurate prediction of PV generation in a short time horizon for network scheduling and dispatching. Over the past two decades, a variety of PV generation forecasts have been proposed [1,2]. Since PV generation in a very short time period (in minutes) can be affected by many factors such as sunshine, wind, and cloud coverage [3], PV generation forecasting is still a challenging task [4].

PV generation is largely dependent on solar irradiance. Thus, the prediction of PV generation can be relatively accurate during sunny days due to the consistent irradiance. In contrast, the prediction of PV generation may not achieve desirable accuracy during overcast, rainy or cloudy days because of considerable fluctuations and irregularities of solar irradiance during these days.

Though the Numerical Weather Prediction (NWP) can facilitate PV generation forecasting over the medium and long-term time horizon [5], it is not applicable to the ultra-short-term forecasting (minutes horizon) [6] due to the fluctuation of local weather conditions at the PV plant in a short time period [7]. Satellite images can provide cloud information; however, they do not contain detailed local cloud distribution above the PV plant. In recent times, the installation of sky-imagers at numerous PV plants has become prevalent. They use ground-based cameras to capture local sky images and can reflect the real-time weather conditions above the PV plant. Sky images can be incorporated into PV generation forecasts to improve prediction accuracy over the ultra-short-term horizon. Therefore, it is necessary to construct a sophisticated machine learning model to understand and reveal the information embedded in the sky images and utilize this information together with historic PV generation data to provide an accurate ultra-short-term PV generation forecasting [8]. Various PV forecasts are summarized below.

1.1. PV generation forecasting using historic PV output data

Other than the inherent randomness, the PV generation time series also exhibits seasonal variations and cyclical fluctuations [9]. Attempts have been made to use statistical methods or machine learning methods to construct PV generation forecasts. In [10], the focus was on outlier detection by using a weighted Gaussian Process Regression (GPR), however, the effectiveness of this approach heavily relies on the availability of high-precision real-time weather data. In [11] an Extreme Learning Machine (ELM) algorithm was adopted. The solar irradiance was also measured and used to compensate for the inaccuracy caused

[*] Corresponding author.
E-mail addresses: shijie.xu@uq.edu.au (S. Xu), zry@ust.hk (R. Zhang), huima@itee.uq.edu.au (H. Ma), chandima@itee.uq.edu.au (C. Ekanayake), yi.cui@usq.edu.au (Y. Cui).

https://doi.org/10.1016/j.solener.2023.112203
Received 6 June 2023; Received in revised form 14 November 2023; Accepted 16 November 2023

by the weather conditions provided by the NWP. Then the model was optimized by both the particle swarm optimization algorithm and the ELM. Similarly, [12] uses the NWP data and a machine learning regression algorithm to improve day-ahead solar irradiance forecasting. Consequently, these two methods exhibit substantial computational complexity and necessitate a significant amount of historical data and accurate NWP information.

Recently, deep learning techniques have been adopted for PV generation prediction. In [13], a multi-layer feed-forward artificial neural network was implemented, the PV plant's geographic information and the power generation of the neighbouring PV panels of the target PV panel were used to infer the cloud distribution, which was then used to predict the PV generation of the target PV panel. In [14], the Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), and Convolutional Neural Network (CNN)-LSTM were implemented for multiple forecast horizons. To improve the data quality of the historic PV generation dataset, in [15] a wavelet packet decomposition was applied to split the PV generation time series into several subsets at different time-frequency scales. These subsets were then fed into four independent LSTM models. [16] utilized the transformer-based structure to do the one-hour ahead PV prediction, the historical PV generation data and NWP data are fed into the model as input. Nevertheless, most of the above methods may lack the capability of capturing the irregular fluctuations of PV generations in the ultra-short-term horizon accurately. Numeric data-based methods, as demonstrated in previous studies [10,11,13], necessitate a substantial amount of additional data and involve significant computational complexity. Furthermore, some of the methods above are primarily applicable to longer forecast horizons, which is due to the lack of real-time weather information. As an easily accessible real-time weather information source, cloud images have been effectively employed to enhance the accuracy of PV generation forecasts.

1.2. Image-based analysis and forecasting

Both satellite and sky images can provide information regarding the local weather conditions. In addition, an image sequence may reveal hidden spatio-temporal relations, which could be used to infer future local weather conditions.

1.2.1. Satellite image-based analysis and forecasting

A satellite covers a wide observation area and can provide high-resolution images of clouds above the solar farm. Numerous solar irradiance and PV generation forecasts using satellite images have been developed. [17] employed a combination of traditional methods, such as the optical flow algorithm and LSTM to catch the hidden spatial features in satellite images. However, traditional physical models may struggle to accurately capture the intricate and complex flow conditions of clouds. In [18] the irradiance estimated from satellite images was fed into a deep neural network (DNN) with convolutional layers to predict the future global horizontal irradiance (GHI) directly. In [19] the spatio-temporal correlations between the target PV plants and its adjacent PV plants were considered, and the PV data from neighbouring plants was utilized to improve the forecasting accuracy of the target PV plant. However, in a satellite image, only a small portion might be of interest for solar irradiance forecast. In [20], an algorithm based on the attention mechanism (AM) was proposed to determine the regions of interest in images. An encoder–decoder structure was then used for estimating irradiance. In [21] the authors analysed both spatial and temporal features of satellite images with a hybrid model of three-dimensional (3D)-CNN and LSTM. The ground GHI records were mixed with a one-dimensional (1D)-CNN to extract the high-dimension feature. 28 An LSTM was applied to estimate the next hidden state, which is used for fusion and generating the prediction. [18–21] all deal with the hours ahead forecast horizon, primarily due to the limited resolution of satellite images they rely upon. For PV generation fluctuation and dispatch forecasts in the minutes ahead horizon, the aforementioned methods provide limited assistance. In [22], the region of interest mechanism was implemented first. Then the U-Net (a variant of the CNN for image segmentation) was used to extract features from the satellite images, which were then combined with NWP data and historical PV generation data. Finally, these data were fed into an encoder–decoder structure based on LSTM with AM. The method proposed in [22] attained better performance because of the inclusion of more exogenous information. However, the small size of the region encompassing a solar farm in comparison to satellite images often leads to an inefficient utilization of storage resources. In [23] the authors focused on various forecast horizons and both sky images and satellite images were utilized. The statistical methods were implemented to predict GHI and direct normal irradiance. Despite the utilization of sky cameras, the effectiveness of the method remains limited when dealing with forecast horizons shorter than 30 min.

1.2.2. Sky image-based analysis and forecasting

The stochastic and non-linear nature of weather conditions has more influence on the performance of ultra-short-term forecasts. In the study by [24], ResNet was used to predict PV generation in the upcoming five to ten minutes. The temporal features of sky images were represented as stacked red channels. On the other hand, in [25], mathematical methods were utilized to extract spatial features and fed them into an LSTM model. However, the absence of spatial or temporal feature analysis significantly hinders the methods' ability to achieve high levels of accuracy in PV generation prediction. Considering both of them, [26] designed two CNN-based structures to handle the sequences of sky images, the first model conducted a two-dimensional (2D) convolution on stacked sky images (the model is denoted as SCNN), while a 3D-CNN directly conducted 3D convolution on the sky image sequence. The result showed that convolutional operation did not perform well over channels and the 2D convolutional structure was better than the 3D structure. This implies that the traditional CNN structure may not be able to provide approximate temporal information. In [3], the authors considered both spatial and temporal features. A CNN and a residual structure from ResNet were used to avoid the vanishing gradients. The consecutive two-time step sky images were fed into the convolutional module separately to learn the rate of change between sky images. A double route structure was designed to catch the spatio-temporal features of sky images. The spatial feature from sky images, the temporal feature from historical data, and the exogenous data were integrated to predict PV generation. In [27] the authors used GHI data-based LSTM and sky image-based CNN auto-encoder (AE) to make predictions with both spatial and temporal dimensions. The extracted image feature was fused with the compressed historical GHI. The method proposed in [28] was also implemented with 2-D and 3-D CNN-AE structures. In the case of [3,27,28], CNN is applied for spatial feature extraction. However, the local vision limitation of CNN in shallow layers restricts its ability of PV forecasting, there is still potential for enhancing the model's capability to handle spatial features. Similar to the approach employed in [17], traditional methods are utilized on sky images in advance of the neural network component in [29]. Still, traditional physical models cannot achieve a significant improvement than CNN-based models. In [30] both the spatial and temporal features of the sky images were considered. PhyDNet and ConvLSTM were used in parallel. PhyDNet focused on the physical dynamics of the residual factors in the sky images. Then, the extracted features were mixed and sent into a CNN decoder for future sky image prediction, and an MLP decoder was used for future solar irradiance forecasting. The image generation-based method contains over-detailed hidden features within the inner layer, potentially affecting the performance of the model. A Vision Transformer (ViT)-based framework was employed in [31], and the NWP data, estimated clear sky irradiance and sky images are fed into the model. However, the exogenous data are directly combined

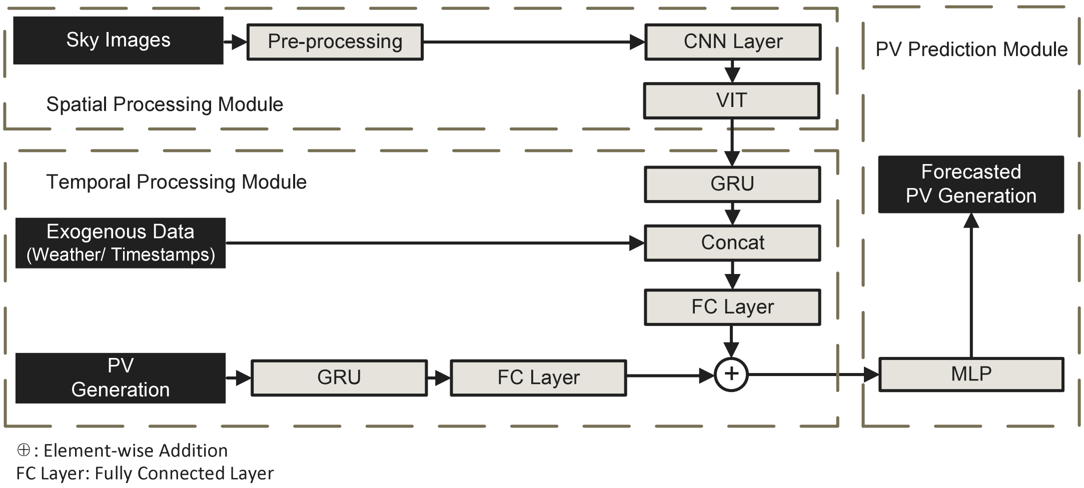

Fig. 1. Overall structure of the proposed PV forecasting framework.

with the embedded sky images, which may cause overshadowing or neglecting the relatively limited exogenous information. Additionally, this framework includes two consecutive self-attention transformer-based structures for both spatial and temporal features. This may potentially lead to some extents of ambiguities in the time flow. The authors in [32] utilized transformer-based structure on the PV generation data and the extracted optical flow maps to perform the 10 to 30 min ahead of multi-step forecasting. The attention mechanism facilitates achieving the desired performance. However, this study heavily relies on the high-resolution historical PV generation series but does not sufficiently utilize the hidden features extracted from the sky images due to the relatively lower resolutions of the sky images.

However, most of the above deep learning models utilize historical PV generation data to capture the temporal features, which may not contain sufficient information regarding solar irradiance fluctuation in the ultra-short period. On the other hand, in the existing models using sky images, a temporal-specialized deep learning mechanism has not been established to catch hidden features related to spatial feature flow within sequences of sky images. Moreover, the commonly used spatial feature extraction model could be improved. This paper addresses the above two issues in a unified framework. The temporal-specialized mechanism is researched and implemented in our model, and a high-performance spatial feature extraction mechanism is also included.

1.3. Contributions

This paper aims to achieve a high accuracy in ultra-short-term PV generation forecasting. We have made the following contributions.

1. A sky image-based spatio-temporal PV forecasting framework is proposed. By integrating a GRU-based structure following static spatial analysis, our proposed framework enables accurate capture of the temporal flow information inherent in sequential sky images. This capability facilitates improved forecasting of PV generation 2110 fluctuations.
2. A novel spatial feature processing module is proposed for sky image spatial analysis. The combined CNN and ViT structure is designed to help the model catch the global vision of sky images for static spatial features.
3. A double-model structure is built to handle the sky images and exogenous information separately. The independent processing of exogenous information allows for a more stable estimation of clear-sky irradiance compared to using historical PV generation data. This approach helps prevent the occurrence of outliers and enhances the reliability of the estimated clear-sky irradiance in our forecasting framework.

2. Problem formulation

In the proposed PV generation forecasting framework, we utilize sky images (SIs) I_H with timestamps $H = \{h-l, h-l+1, \ldots, h\}$ as the main input, where l is the length of input SI sequence. The same timestamps are adopted for the following data: (1) PV generation G_H; (2) the time of a day T_H, which is used to indicate a rough value of PV generation at that time in a clear day as a reference; and (3) numerical weather prediction W_D, where the subscript D denotes the one-day interval.

Over the forecast horizon Δ, the prediction (output) $\hat{P}_{h+\Delta}$ is generated from the forecasting model. The whole proposed framework could be represented as a nonlinear mapping function from the input data to the predicted PV output, which is denoted as M. Thus, the problem can be formulated as:

$$\hat{P}_{h+\Delta} = M(I_H, G_H, T_H, W_D) \tag{1}$$

3. Methodology

3.1. Overall structure

The overall structure of the framework is shown in Fig. 1. The sky image sequence I_H and PV generation data G_H are processed separately. Since the sky image sequence comprises both spatial and temporal features, a single-feature neural network is not sufficient for extracting the hidden features embedded in the image. Therefore, in the sky image processing route, a model combining a convolutional layer, a ViT encoder and a GRU encoder is implemented. A second GRU model is implemented for extracting the hidden features from the PV generation data. The features extracted from the sky images are treated as fundamental information, while the features extracted from the PV generation data are treated as extra information. 21 Then, these two features are element-wise added as the mixture information. Finally, a residual linear model is used to handle the mixture of hidden information and generate the final estimation of the PV generation over the time horizon.

3.2. Spatial feature processing module

In ultra-short-term PV generation forecasting, cloud 28 dynamic is the most significant influencing factor. The clouds may suddenly block the direct sunlight on the PV panels or move away from the sky above the PV panels. This can lead to a rapid increase or decrease in solar irradiance. The sky image series can capture such solar irradiance changes. Thus, in the first step, we adopt ViT to extract the hidden spatial features from the sky images. Compared to the traditional

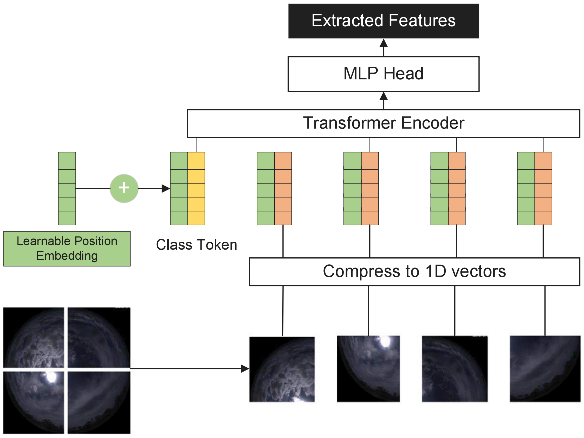

Fig. 2. ViT-based embedding structure.

cloud detection methods and CNN-based models, ViT can attain better performance.

ViT [33] splits the original image into numerous small image bucks and focuses on the most important part subsequently extracting useful hidden features with its self-attention mechanism. Compared to the CNN-based sky image processing structures, the above approach in ViT makes the capture of the global information from the images in the shallow layers better [34]. In the PV generation forecasting task, the global information obtained from the entire sky image series over a certain time period could better represent the weather trends. Thus, ViT-based structures are more suitable for sky image processing than CNN-based structures. Moreover, PV generation forecasting is a fixed-length regression task, which is different from the natural language processing task. Thus, in the encoder–decoder mechanism of the vanilla transformer structure, the decoder is not required and only the encoder is required for feature extraction. Additionally, [35] shows the convolutional layer at the early stage could improve the robustness and converge speed of the transformer model. Thus, a convolutional layer is combined before the ViT model to improve its performance.

In the spatial feature encoder (Fig. 2), the transformer layer is built based on the ViT-based structure. Because the vanilla transformer model receives a one-dimensional image sequence as the input, the input sky image $I \in R^{H \times W \times C}$ is firstly split and flattened to a 2D array $I_A \in R^{N \times (P_H \times P_W \times C)}$. H is the height of the input image, W is its width, C is the number of channels, P_H and P_W refer to the height and width of a patch, and N is the number of split patches. After patch embedding, the vectors are concatenated with a 'class' vector similar to the 'class' token in [36], which represents the compressed status of patches and avoids the preference for specific patches. The position embedding E_p is added with all the patch vectors to contain the positional information of the patches in the original image and their relativity.

Then, the patch vectors are sent into a vanilla transformer encoder. The structure of the transformer encoder and the MLP is shown in Fig. 3. L is the number of encoder layers, all the 'Norm' blocks are Layernorm as proposed in [38], and the residual connections are implemented twice in one block to avoid the gradient vanishing problem. The spatial features in static sky images will be extracted. This module could be formulated as:

$$F_s = V(I_H; \theta_v) \tag{2}$$

where F_s is the hidden spatial features, V denotes the ViT-based spatial feature processing module, and θ_v is the parameters of this module. The high-dimensional spatial features are extracted by the above non-linear mapping model and used in the temporal analysis. The spatial features are expressed in the form of matrices, which consist of the weighted values of the influential regions (e.g., sun, cloud-covered areas) within the sky images. These matrices represent the fundamental characteristics extracted from the sky images, including the relative position of the sun, cloud distribution, irradiance intensity, and other environmental information. The core of this module is a multi-head self-attention structure [37] as shown in Fig. 4.

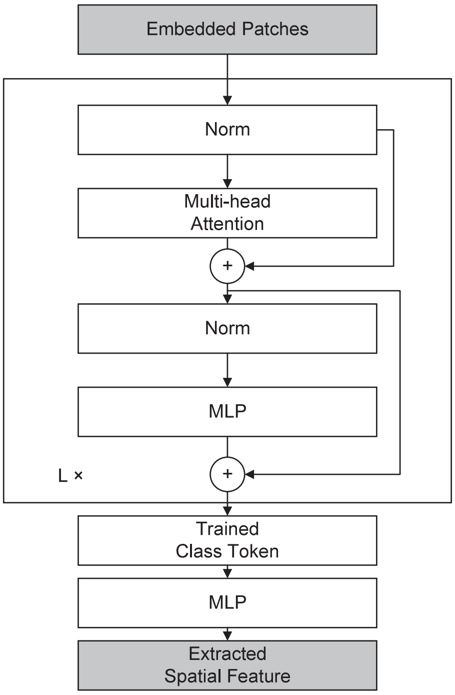

Fig. 3. The structure of the transformer encoder and MLP head [37].

The above attention mechanism is designed for the transformer-based structure and 28 is known as "Scaled Dot-Product Attention". It calculates a query and a set of key–value pairs to a weighted output. The weighted values are from the corresponding index between the query and its key. With the trainable and strongly correlated weights, the basic attention structure is used for capturing the latent relationship between the query and the key–value pairs. This self-attention mechanism generates the query, key, and value vectors using the same input. It can successfully extract both the independent hidden features and internal latent correlation from the different segments belonging to one input. Accordingly, the query matrix Q, the key matrix K, and the value matrix V are calculated by multiplying the embedded input with the weight matrix obtained during the training process. The attention mechanism can be formulated as:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{3}$$

where d_k refers to the dimension of the key vectors. Then, the multi-head attention is applied to the transformer structure. Thus the model can use multiple attention heads to capture various aspects of the input with one head corresponding to one aspect. The attention vectors from different heads are then concatenated and mapped to the final output. The formula of this mechanism can be represented as:

$$\begin{aligned} MultiHead(Q, K, V) = Concat(&Attention_1(Q, K, V), \\ &..., Attention_h(Q, K, V)) \end{aligned} \tag{4}$$

where $Attention_i$ is the output of attention head i. After that, a 'class' vector completes the information exchange through the attention process. The vector will be selected to represent all the hidden features in all the attention spaces. Through an MLP, F_s will be generated from the 'class' vector as an array with the required dimensions. Fig. 5 shows

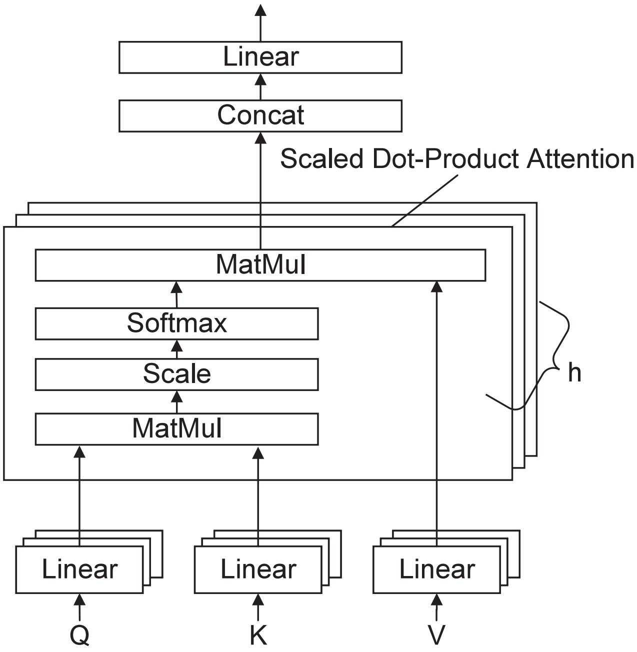

Fig. 4. The multi-head self-attention structure [37].

the attention maps of a sky image sequence (10-minute interval), 1112 11 which serves as the comprehensive summation of all heads across all layers within the entirety of the framework. These attention maps are generated through the computation of the average weight derived from all heads. They could represent the regions that the whole spatial feature processing module will be interested in. In Fig. 5, the essential influential areas (highlighted in warmer colours) for PV generation, such as the sun and nearby clouds, will be assigned higher weights during feature extraction, it can be observed that the highlighted areas flow as the clouds and sun move. Thus, the majority of the spatial feature matrices F_s will encompass weighted values corresponding to these highlighted areas.

3.3. Temporal feature processing module

Using the spatial feature processing module presented in the previous section, the hidden high-dimensional features such as cloud distribution and sunlight strength are extracted from the static sky images. However, these hidden static features could only consist of the instant status of the weather. The temporal features should also be included for completing the forecasting task.

Considering the consecutive temporal distribution in the sky images, hidden temporal features exist between the extracted spatial feature vectors. Since recurrent neural networks (RNNs) (e.g. GRU and LSTM) can capture context within consecutive time series, we implement a GRU structure encoder to extract and compress the features from the time series. Even though the self-attention layers can be used for temporal feature extraction, the RNNs are more appropriate for this task. This is because both the spatial–temporal features and the PV generation data are one-directional. So, the one-directional GRU could effectively capture the temporal relationships along the same time flow direction. The characteristics at one timestamp will influence the characteristics of the subsequent timestamps. The closer the two data points (along the time stamps), the stronger the correlation exists. However, in the self-attention layers, the embedding mechanism is employed to allocate the distances between data points. It cannot follow the specific time flow direction effectively as the GRU could do. Therefore, the computational complexity of self-attention layers is higher than that of the GRU. Compared to the LSTM structure [39], the GRU structure requires fewer parameters but can achieve similar performance. In the temporal processing module, the extracted spatial feature sequences and PV historical generation sequences are fed into two separate GRU models. The structure of GRU is shown in Fig. 6.

The GRU model could be formulated as:

$$\begin{aligned} z_t &= \sigma_g(W_z x_t + U_z h_{t-1} + b_z) \\ r_t &= \sigma_g(W_r x_t + U_r h_{t-1} + b_r) \\ \hat{h}_t &= \tanh(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h) \\ h_t &= z_t \odot \hat{h}_t + (1 - z_t) \odot h_{t-1} \end{aligned} \tag{5}$$

where x_t and h_t are the input and output vectors at time step t respectively. z_t and r_t are the update and reset gates vectors. $\hat{h}_t$ is the current memory content. W and U are the parameter matrices, b is the bias vector. σ is the sigmoid function, and $\odot$ is the Hadamard product.

GRU is a variant of the basic recurrent neural networks [40]. It avoids the gradient vanishing problem and incorporates the gate operating mechanisms to preserve long-term memories while preventing short-term memories from covering long-term memories. The update gate z is used to decide which part of the previous hidden states should be kept to the next step while the reset gate r is used to determine which part of the previous hidden states should be discarded. The useful information from the last step $r_t \odot h_{t-1}$ will be added with the weighted current input and constrained between 0 to 1 by a $tanh$ function. This calculation result is the current memory content $\hat{h}_t$. Finally, the update gate vector will be element-wise multiplied with the current memory content as $z_t \odot \hat{h}_t$, the previous hidden state will be element-wise multiplied with $1 - z_t$ as $(1 - z_t) \odot h_{t-1}$. The above two parts are summed together to generate the output vector h_t.

By recurrently feeding the extracted spatial features of SIs from the previous spatial feature processing module to the time dimension, the final GRU cell can generate the encoded vector h_{final}. For the spatial feature input, its corresponding output contains both the spatial features and temporal features that are extracted by two consecutive spatial and temporal specialized analysis modules.

In Fig. 7, the input spatial feature series F_s^{in} and historical PV generation time series G_h^{in} of window n to GRU encoders are denoted as:

$$\begin{aligned} F_s^{in} &= (F_s^{t-i-n+1}, F_s^{t-i-n+2}, \ldots, F_s^{t-i}) \\ G_h^{in} &= (G_h^{t-i-n+1}, G_h^{t-i-n+2}, \ldots, G_h^{t-i}) \end{aligned} \tag{6}$$

where t is the target prediction timestamp, i is the prediction interval, and n is the length of the input series.

As shown in Fig. 7, 28 the spatio-temporal features F_{st} are generated by the GRU Encoder One with the extracted spatial features fed in. The spatio-temporal features are represented as matrices. They will be high-dimensional data that include the sun's trajectory, cloud velocity and direction, cloud deformation and other hidden dynamic spatial features inherent in sky images. At the same time, the historical PV generation data G_h is fed into the GRU Encoder Two to generate the time series-based auxiliary PV generation feature F_g. Through the whole temporal module we can have:

$$\begin{aligned} F_{st} &= GRU_1(F_s^{in}) \\ F_g &= GRU_2(G_h^{in}) \end{aligned} \tag{7}$$

3.4. PV generation prediction module

The decoding and prediction processing after the temporal feature processing is also shown in Fig. 7, where n is the number of time steps for each input, and 'FC Layer' means the linear fully connected layer.

After going through the temporal processing module, the hidden spatio-temporal features are all extracted and they are the conclusive expression of the input SIs. To locate the precise weather situation, numeric exogenous data is concatenated with the spatio-temporal features from the sky images. The relative time E_{time}, rainfall E_r, solar irradiance

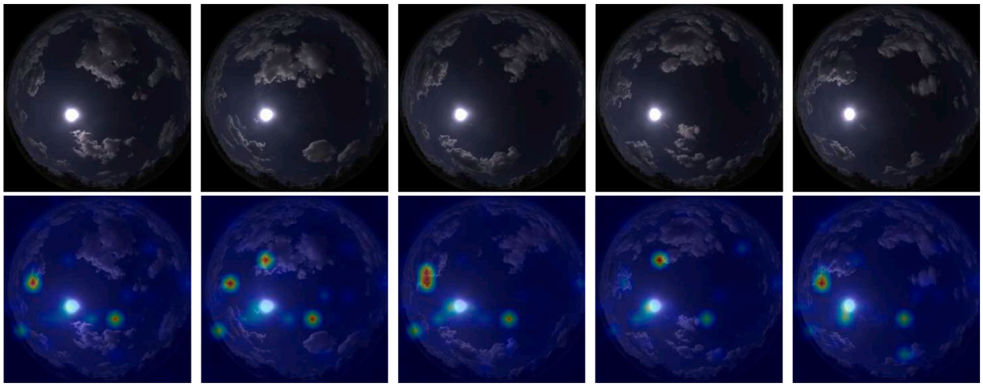

Fig. 5. The attention maps on a cloudy sky image series (10-minute interval, from left to right in chronological order).

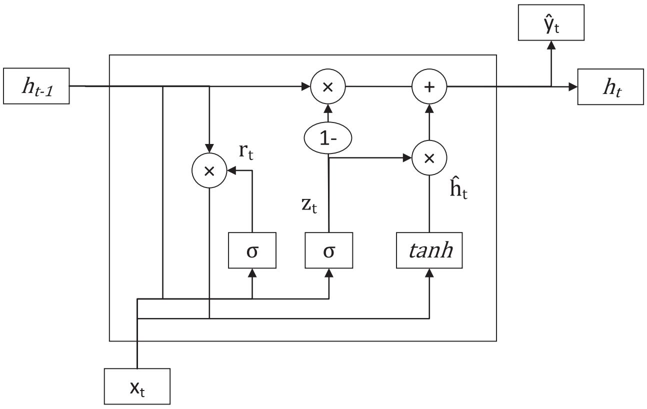

Fig. 6. Default GRU structure.

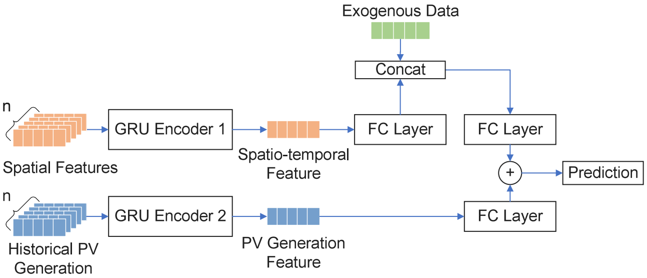

Fig. 7. Temporal processing and PV generation prediction structure.

E_s, and temperature records E_{temp} are min–max scaled to [0, 1]. The combination of the exogenous data F_e and feature-wise scaling for each exogenous feature f_e are:

$$F_e = [E_{time}, E_r, E_s, E_{temp}]$$
$$f_e^{minmax} = \frac{f_e - min(f_e)}{max(f_e) - min(f_e)} \tag{8}$$

F_e will be concatenated with F_{st}. The numeric F_e represents the standard solar irradiance similar to the clear-sky model, which could help the model locate the rough range of the PV generation.

Then, two fully connected layers are set to compress the vectors to the output dimension. In order to compare the performance of sky image-based forecasts with and without using the historical PV generation data, an individually trained GRU module is used to extract the PV generation feature. After all, the exogenous and spatio-temporal combined feature vectors are added for the final prediction P_t.

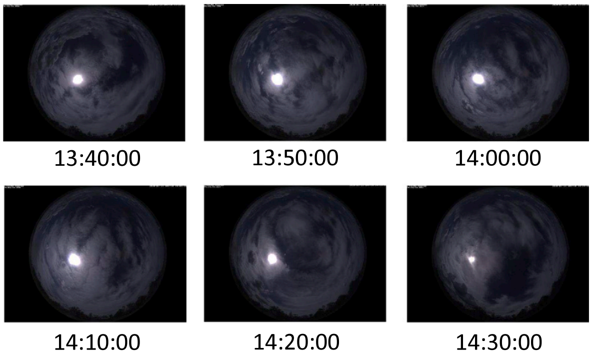

Fig. 8. Sample sky images of one hour at noon on January 23, 2020.

4. Dataset and experiments

4.1. Datasets

The data regarding PV generation and sky images were acquired from a solar farm situated at The University of Queensland's Gatton campus in Australia, possessing a peak capacity of 3.275 MW. Two MOBITIX hemispheric cameras (Q25) have been installed to take real-time images of the sky above the solar farm. The sky image has 2408 × 1536 pixels and RGB channels and the time resolution of the images is 10 s. PV generation data is with a resolution of one minute. In the following case studies, the dataset was selected for the time period from January 23rd, 2020 to May 6th, 2020. The total size of the sky images is 61.9 GB. The sample sky images in one hour are shown in Fig. 8 (10-minute interval). Based on the historical irradiance record of the solar farm in this paper, it is observed that the number of days with irradiance levels exceeding the average account for 50.96% of the total days, while those with irradiance levels below the average constitute 49.04% of the total days. 23 Moreover, the variance of the difference between the real data and the clear-sky estimate model is also considered, which could represent the strength of fluctuation. The percentage of days with higher fluctuations than the average is 49.51%, which is close to the number of days with milder fluctuations. This implies the dataset is not dominated by any single specific weather condition. So, the trained model has the capability to generalize well across different weather conditions. Part of the dataset, which includes the sky images and PV generation data, has been published in [41].

The weather datasets are collected from the Bureau of Meteorology, Australia. The weather station is located in the solar farm. The weather data consists of rainfall, solar irradiance, and temperature. The resolution of the numeric weather data is one day. The weather data is based on the Australian Community Climate and Earth-System Simulator (ACCESS) weather model and collected weather station data which are both provided by the Bureau of Meteorology[42].

Table 1
Parameters of the training stage.

Hyper-parameter	Configuration
Batch size	8
Learning rate	0.00001
Epochs	40
Early stop epoch	4
Length of inputs (mins)	5
Length of output (min)	1
Transformer dropout ratio	0.1
Embedding dropout ratio	0.1

4.2. Pre-processing

Since the raw sky images are with over-high resolution and contain unneeded wrapping, they are cropped to 1500 × 1500 at centroid and resized to 258 × 258 to fit the convolution layer before the ViT-based model. For the baseline method, the sky images were resized to 256 × 256 to fit the original input size of the model. Then, the RGB values from [0, 255] were scaled to [0, 1] and normalized to avoid the gradient vanishing problem.The standardization is implemented on the pixel values to reduce the impact of noise and outliers. The pixel values P_i were normalized with the mean P_{mean} and standard deviation P_{std} of the entire sky image dataset:

$$P_{norm} = \frac{P_i - P_{mean}}{P_{std}} \tag{9}$$

The various ranges of the PV generation data and weather data are both rescaled to [0, 1] by a min–max scaling:

$$N_{minmax} = \frac{N_i - N_{min}}{N_{max} - N_{min}} \tag{10}$$

where N_i is the numeric data of a time series, N_{min} and N_{max} are the minimum and maximum values of the time series.

4.3. Experiment setting

Experiments are conducted on the high-performance computing (HPC) GPU nodes. Each HPC node contains two Intel Xeon Gold 6132 CPUs, four NVIDIA SXM2 V100 Accelerator Units, and 384 GB of RAM. All the models including the baseline methods were developed with PyTorch 1.9.1 and Python 3.8.11. The whole dataset was randomly split into three parts: a training set, a validation set, and a testing set. The ratio of the three datasets is 6:2:2. To avoid over-fitting, only the training set was used for training, the validation set was fed into the model after each training epoch to track the immediate performance of the model, and the testing stage was run separately. Furthermore, the dropout is implemented on the transformer model and embedding stage to prevent our model and embedding result from overfitting on the training set. For the SI input I_H, the chosen time length is 5 min. To save the computational resource, the resolution of input sky images is reduced and tested. (The comparison of different resolutions will be shown in section 4.5) 20-second interval is chosen to be the input resolution, this implies that there are 15 images with this interval fed in as one input. The parameters of the training stage are shown in Table 1.

The hyper-parameters are tuned across preliminary experiments. Considering the high GPU memory consumption of the ViT-based structure, we set our training and testing batch size to eight to avoid out-of-memory problems. With sufficient training data and epochs, our initial learning rate was set to 0.00001 to prevent over-fast convergence during the training stage and ensure optimal model generation. The transformer and embedding dropout ratio are both set to 0.1 to avoid overfitting and preserve the most learned information at the training stage. Furthermore, to avoid overfitting and save computational resources at the same time, the training procedure is designed to stop either upon observing a deterioration or only a marginal improvement of validation loss for four consecutive epochs. The structural parameters of our model are shown in Table 2.

Table 2
Structural parameters of our model.

Parameter	Configuration
Convolution layer	
Number of kernels	3
Kernel size	3×3
ViT module	
Patch size	16×16
Dimensions of embedded vector	1024
Number of heads	4
Dimensions of head	64
Transformer layers	4
Dimensions of MLP	2048
Dimensions of the output layer	256
GRU module	
GRU encoder 1 hidden state	256
GRU encoder 1 layers	4
FC layer after GRU encoder 1	(4×256, 256)
FC layer after concatenate	(256+4, 1)
GRU encoder 2 hidden state	2
GRU encoder 2 layers	4
FC layer after GRU encoder 2	(2×4, 1)

Through extensive preliminary experiments using smaller training sets and different model parameters, we have identified the optimal hyper-parameters in Table 2. These parameters can ensure the best performance over the sky image dataset and avoid unnecessary structure redundancies, such as overlarge transformers and redundant GRU layers.

To reduce the computational resource consumption, the Adam algorithm [43] is used as the optimizer of our model, which could automatically adjust the learning rate. For the regression task, the Mean Square Error (MSE) loss is chosen as the loss function to optimize the model output.

4.4. Baseline methods and evaluation metrics

4.4.1. Baseline methods

Four baseline methods were implemented: ResNet [3], SCNN [26], ConvLSTM [30] and the smart persistence model. The smart persistence model assumes the relative PV generation, which is the ratio of the PV generation to the estimated clear sky PV generation, stays constant. In this algorithm, the relative PV generation G_t/E_t at the current time step t is used as the predicted PV generation G_{t+1}/E_{t+1} at the next time step. The estimated clear sky PV generation E_t comes from the Ineichen and Perez clear sky model [44]. The set of baseline models comprises traditional mathematical methods, spatial-based methods, and spatio-temporal hybrid methods. Our model is compared with these four baseline models to demonstrate the advantages in the spatial feature extraction and spatio-temporal feature learning ability of our structure over the traditional temporal-considered CNN model.

4.4.2. Evaluation metrics

The performance of all the models was evaluated with four metrics: Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and R-squared (R^2):

$$MAE = \frac{1}{N}\sum_{i=1}^{N} |y_i - \hat{y}_i| \tag{11}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (y_i - \hat{y}_i)^2} \tag{12}$$

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} = 1 - \frac{\sum_{i=1}^{N} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{N} (y_i - \bar{y}_i)^2} \tag{13}$$

Table 3
MAE compared to the persistence model.

Forecast horizon/%	5 mins	10 mins	15 mins	Average
SCNN	−10.63	−3.89	8.85	−1.89
ResNet	−31.16	−13.53	−1.31	−15.34
ConvLSTM	−28.74	−10.00	0.60	−12.71
Our model without historical PV data	37.74	**42.29**	**37.53**	**39.19**
Our model with historical PV data	**38.32**	40.63	37.38	38.78

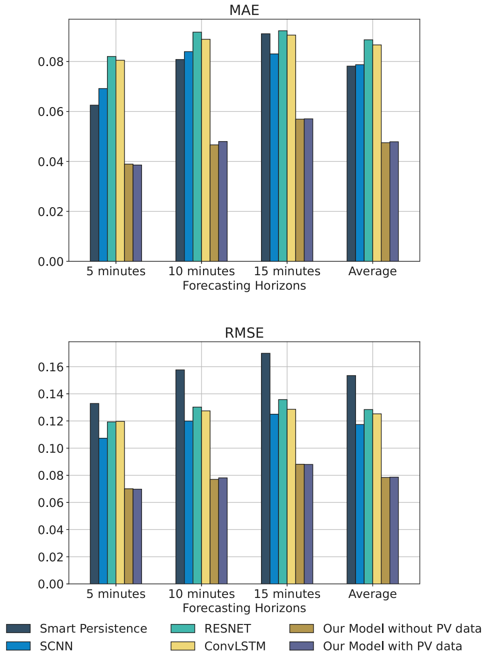

Fig. 9. Comparison of error metrics on different time horizons.

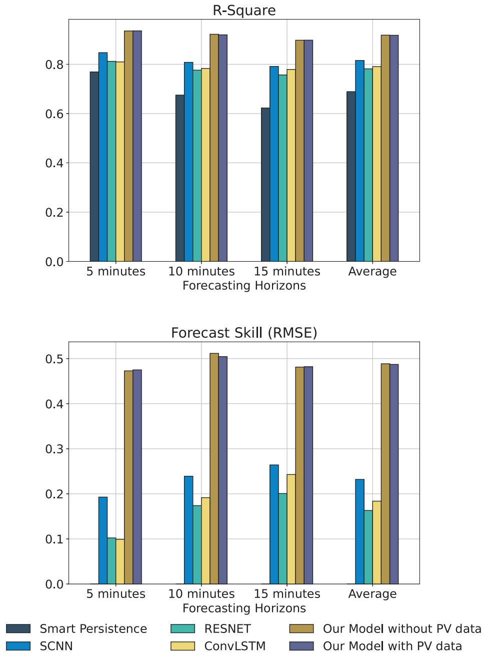

Fig. 10. Comparison of R-square and FS metrics on different time horizons.

where y_i is the ground truth, $\hat{y}_i$ is the prediction, and $\bar{y}_i$ is the mean of y_i. Many papers use normalized RMSE (nRMSE) as the metric for comparison. In our case, the predicted PV generation value is min–max scaled between 0 to 1, thus, the RMSE value is equivalent to the nRMSE.

$$nRMSE = \frac{RMSE}{y_{max} - y_{min}} = RMSE \tag{14}$$

2515The Forecast Skills (FS) was computed to indicate the improvement of the model from the baseline model:

$$FS = 1 - \frac{RMSE}{RMSE_{baseline}} \tag{15}$$

4.5. Experiment results

Two versions of our model, i.e. the sky-image-based forecast with and without using the historical PV generation data are tested. Such an arrangement is for the comparison of the effect of the historical PV generation data on the performance of our sky-image-based forecasting model. All four baseline models and our two models are trained on the same randomly split training set with different forecast horizons. The deep learning models were all trained on the GPU for the accelerated computation, the smart persistence model was implemented on the CPU. The ResNet is the ResNet50 version, and the ConvLSTM has two layers. Our model with historical PV generation receives both sky images and PV generation data as input, the ResNet, ConvLSTM, and SCNN, and our model without historical PV generation receive only the sky images as input, and the persistence model only requires the historical PV generation data as input.

4.5.1. Evaluation on the testing set

2510 2530Firstly, the models are evaluated on the whole testing set. Comparisons are made with four different metrics on the 5-minute, 10-minute, and 15-minute forecast horizons. For easier comparison, MAE and RMSE are shown in Fig. 9 while R^2 and Forecast Skill are shown in Fig. 10.

It can be seen from Fig. 9 that both our two models perform better than the baseline models for the 5-minute, 10-minute, and 15-minute time horizon. With the increase of the forecast time horizon, the error of the persistence model obviously increases. 25 The higher MAE and lower RMSE of the CNN-based baseline models represent that although the CNN-based models do not follow the regular PV generation curve as stably as the persistence model, they have a small number of significant errors, especially during periods of fluctuations.

The MAE values of our models are consistently lower than that of all the baseline models. As presented in Table 3, all the CNN-based baseline models perform even worse than the persistence model in the 5- and 10-minute horizon. In contrast, our models (with and without using historic PV generation data) consistently perform better than the persistence model (37 ∼ 42% improvement in performance). Moreover, for the 15-minute forecast horizon, our two models achieve around 37% improvements. Our model with historical data performs best in the 5-minute forecast horizon. Our model without historical data performs better in the 10-minute and 15-minute horizons.

The RMSE-based Forecast Skills (Eq. (15)) scores are shown in Fig. 10. The higher FS score indicates the model achieves a higher improvement with respect to the persistence model. It is observed in Fig. 10 that all five deep-learning models have positive FS scores. Specifically, for the 5-minute forecast horizon, our two models exhibit the most significant improvement compared to the other baseline

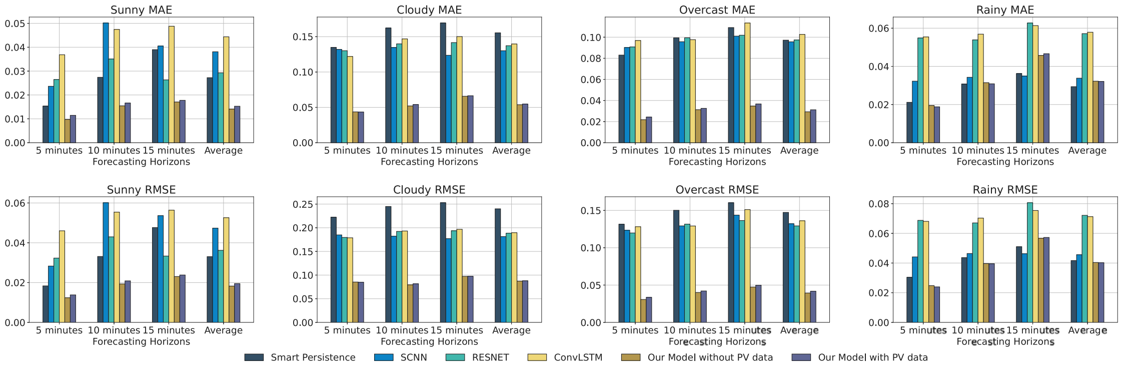

Fig. 11. Comparison of MAE and RMSE under different weather conditions.

models. Moreover, our model with historical PV data shows similar performance compared to our model without using historical PV data.

In Fig. 10, the R^2 scores are also presented. A higher R^2 score indicates the model is better fitted to the mapping between the input and output. It can be seen that the three deep learning models, which use the sky images as input, all achieve higher R^2 scores with respect to that of the persistence model. This implies the sky images contain rich information with respect to PV generation and can help to improve the forecasting accuracy of PV generation. Moreover, our models perform better than the other baseline models on all three forecast horizons. Therefore, our models could fit the correlation between the input and future PV generation better.

Across the experiments on the complete testing set, though the performance of our models may vary with different forecast horizons, they consistently outperform all baseline models in terms of MAE, RMSE, and R^2. PV generation prediction of solar farms is used for the dispatch operation by the grid operator. However, a significant outlier in the prediction could cause difficulties in dispatch. Among the four metrics utilized in this section, RMSE demonstrates the highest sensitivity to outliers. The lowest RMSE values of our models imply that our models could minimize the possibility of the occurrence of significant outliers, thereby enhancing the reliability of the PV generation predictions.

4.5.2. Evaluation under different weather conditions

Solar irradiance significantly varies under different weather conditions. The easier predicted sunny data could lead to an over-reliable result of the persistent model. It is thus necessary to verify the performance of PV generation forecasts under different weather conditions. The above trained models are tested on four types of days including sunny, cloudy, overcast and rainy days. The results are shown in Fig. 11.

From Fig. 11, we can see that on the sunny day, the persistence model exhibits good performance for the 5-minute forecast horizon. This is because the change of irradiance is insignificant during 5 5-minute period on a sunny day. For the 10-minute and 15-minute horizons, the SCNN model has the worst performance. The reason could be that on the sunny day, the complexity of sky images affects their performance which relies on spatial information. However, our two models both perform well in 10-minute and 15-minute horizons.

On the cloudy day, the average MAE and RMSE values are higher than that of the other weather conditions. The cloudy weather condition leads to large fluctuations in PV generation; in turn it can affect the performance of PV output prediction. 2710 With the considerable fluctuation of solar irradiance, the persistence model could not make a meaningful prediction. Benefiting from the spatial feature analysis on sky images, the three baseline deep-learning models achieve better performance than the persistence model. By incorporating the detailed spatio-temporal feature analysis, our models have outstanding improvements over all the baseline models and achieve the lowest errors. In this condition, the historical data has almost no effect on the accuracy of forecasts. This indicates the main information source is the sky images and the exogenous data, our spatio-temporal oriented structure could catch both types of information well.

The overcast day is similar to the cloudy day. Even under the high randomness of weather, our models still can provide the desired PV output prediction (i.e. the lowest MAE and RMSE in Fig. 11). However, because of the lower average solar irradiance on the overcast day, the distinction between our models and the baseline models is not as significant as observed on the cloudy day.

As depicted in Fig. 11, it is evident that none of the models exhibits satisfactory performance on the rainy day. This is because all sky image-based models are highly dependent on the explainable phenomenon of clouds. However, the sky images could not interpret the sky conditions with the extremely complicated formation and movement of rainy clouds. Moreover, the raindrops could also affect the operation of the optical sky-imagers.

The comparison between our model of 5-minute, 10-minute, and 15-minute forecast horizons, the CNN-based baseline models of the 5-minute forecast horizon, and the ground truth is shown in Fig. 12. From the figure, we can see that the baseline models can roughly recognize the peak and off-peak, but they cannot accurately catch the magnitude of the changes. In contrast, on the 5-minute and 10-minute horizons, our model can precisely capture the trends of drops and rises in PV generation, which fluctuates rapidly in the short time period. This implies our models can help the grid operators to properly plan the dispatch in the ultra-short term.

4.5.3. Saliency map comparison

The above results provided a comprehensive verification of the models' performance. However, the comparison only using the metrics does not clearly present how the ViT-based structure can enhance the spatial feature extraction process. Thus, the saliency maps are utilized to visualize the 'attention' of CNN-based models, which are heat maps that could catch the highest concern areas within the models. The saliency maps are generated by computing the gradient of the pixels of one sample sky image to its output [45]. The saliency maps of the CNN-based models are shown in Fig. 13. From Fig. 13, we can see that the three CNN-based models have various concern patterns. The ResNet pays attention to most of the cloud pixels and focuses on the nearest cloud area to the sun, but the sun is ignored. Moreover, it pays more attention to every pixel than other models, which is limited by the local vision of standard CNN, it ignores the less essential pixels of the whole sky image. In contrast, the SCNN pays the most attention to the sun area but only pays slight attention to the surrounding area. The ConvLSTM also focus more on the cloud areas, but mainly on the edge area, which may be affected by its mixed CNN and RNN structure. Viewing the 'attention' heat map of our ViT-based model Fig. 5, it can be seen that our model pays attention to both the sun and cloud pixels, which are the most influential factors affecting PV generation. This implies that the self-attention mechanism makes a considerable contribution to the model's comprehensive understanding of sky images.

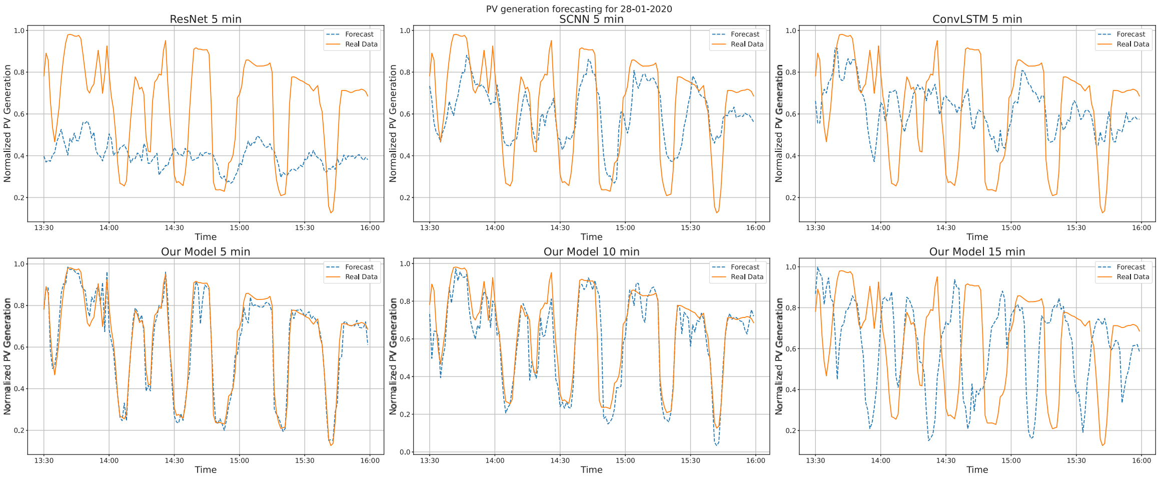

Fig. 12. Comparison of predictions and ground truth.

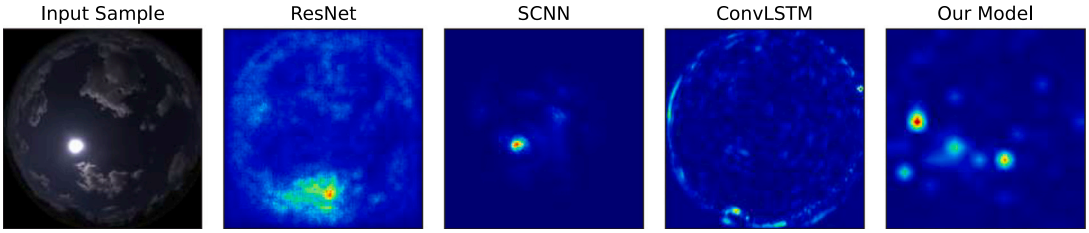

Fig. 13. Comparison of Saliency Maps.

Table 4
Performance comparison of the model with different configurations (input intervals, random seed, with/without exogenous weather data, non-standardized/ standardized PV generation data, GRU-/Transformer-based temporal module).

Forecast horizon					5 mins		10 mins		15 mins	
					MAE	RMSE	MAE	RMSE	MAE	RMSE
Image Interval (s)	Exogenous Data	PV Data	Seed Changed*	Temporal Module						
20	✓	×	no	GRU	0.0389	0.0701	0.0466	0.0770	0.0569	0.0881
60	✓	×	no	GRU	0.0513	0.0843	0.0526	0.0838	0.0553	0.0848
20	×	×	no	GRU	0.0447	0.0779	0.0465	0.0778	0.0584	0.0913
20	✓	×	yes	GRU	0.0440	0.0748	**0.0464**	**0.0757**	**0.0428**	**0.0723**
20	✓	×	no	Trans****	0.0544	0.0875	0.0634	0.0946	0.0616	0.0930
20	✓	Norm**	no	GRU	**0.0386**	**0.0698**	0.0480	0.0781	0.0549	0.0846
20	✓	Std***	no	GRU	**0.0386**	**0.0698**	0.0479	0.0781	0.0567	0.0881

* The random seed for the initialized state of the model.
** Normalized-only: Refer to Eq. (10).
*** Standardized: Refer to Eq. (9).
**** Transformer-based.

4.5.4. *Comparison of different settings and inputs*

1215 1312.5 2410To demonstrate the enhancement achieved by our proposed model, we conducted experiments using various input configurations, and the comparative results for all these configurations are presented in Table 4.

To highlight the impact of the exogenous weather data, the model with only the sky images as input is trained and compared. Table 4 shows that in the 5-minute forecast horizon, the exogenous could bring a notable improvement. In the 10-minute and 15-minute horizons, the effectiveness of exogenous data becomes insignificant, but it could still contribute to error reduction. Overall, the exogenous could enhance the forecasting results, especially in the 5-minute forecast horizon.

In Table 4, we can see that the change of random seed will slightly affect the performance of our model. In the 5-minute forecast horizon, both the MAE and RMSE exhibit marginal increases compared to the default seed. However, in the 10-minute and 15-minute forecast horizons, the performance of the model is slightly improved. Overall, although the training and validation datasets from different seeds can have a minor impact on our models' performance, both the models from default and changed random seeds could provide high-accuracy forecast results. Furthermore, both the models from default and changed random seeds attain desirable performance and outperform all baseline models. The random split of the training set would not significantly affect the stability and reliability of our models.

12 To compare the difference between the one-directional GRU and the self-attention structure for the temporal module, we have trained the Transformer-based temporal module. In Table 4 we can see the performance of the Transformer-based structure is worse than all the GRU-based structures. Also, the execution time for the two modules was compared using 20 data points. The GRU block spends 0.261 s in computation, and the transformer-based block spends 0.364 s, which leads to a higher computational burden.

To enhance the influence of the historical PV generation data in the input, the standardized PV data (Eq. (9)) instead of the normalized-only

PV data (Eq. (10)) is adopted in the input along with sky images and weather data. However, the result (Table 4) shows that PV generation data after standardization operation does not lead to any improvement of the model's performance.

In the proposed framework, instead of using the historical PV data as the only support information, we used the exogenous data to help constrain the forecasting output in an estimated range, which could play a similar role as the historical PV data. Moreover, our spatio-temporal structure can precisely capture the sufficient temporal features of PV generation, which historical PV data typically provide, thus, the standardized historical PV generation is not compulsory required for the model as input.

5. Conclusion

This paper proposed a vision transformer-based ultra-short-term PV generation forecasting framework that focuses on extracting and utilizing spatial and temporal features embedded in sky images. The original sky image sequences are fed into the ViT to obtain the static spatial features and then analysed by the GRU model. On the other hand, the time stamps and environmental information (rainfall, solar irradiance and temperature records) are compressed, scaled and concatenated with the spatio-temporal features obtained from the ViT and GRU models. The fused data is finally used in a fully connected layer decoder to predict PV generation. Additionally, the historical PV generation data is set as an optional input for comparison purposes. Our model outperforms the baseline models and existing deep learning models under different weather conditions for different time horizons. In future work, we will explore the following potential research directions that are limited by current models: (1) Multi-step PV generation forecasting. (2) Precise PV forecasting under rainy conditions. (3) Probabilistic PV forecasting combined with load forecasting for dispatch. (4) Day-block shuffle model on an all-weather condition and full-year-long sky image dataset.

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.

References

[1] J. Khan, M.H. Arsalan, Solar power technologies for sustainable electricity generation – A review, Renew. Sustain. Energy Rev. 55 (2016) 414–425, http://dx.doi.org/10.1016/j.rser.2015.10.135.

[2] V. Le Guen, N. Thome, A deep physical model for solar irradiance forecasting with fisheye images, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2020, pp. 630–631.

[3] R. Zhang, H. Ma, T.K. Saha, X. Zhou, Photovoltaic nowcasting with bi-level spatio-temporal analysis incorporating sky images, IEEE Trans. Sustain. Energy 12 (3) (2021) 1766–1776, http://dx.doi.org/10.1109/TSTE.2021.3064326.

[4] R. Ahmed, V. Sreeram, Y. Mishra, M. Arif, A review and evaluation of the state-of-the-art in PV solar power forecasting: Techniques and optimization, Renew. Sustain. Energy Rev. 124 (2020) 109792.

[5] D.P. Larson, L. Nonnenmacher, C.F. Coimbra, Day-ahead forecasting of solar power output from photovoltaic plants in the American Southwest, Renew. Energ. 91 (2016) 11–20.

[6] M. Diagne, M. David, P. Lauret, J. Boland, N. Schmutz, Review of solar irradiance forecasting methods and a proposition for small-scale insular grids, Renew. Sustain. Energy Rev. 27 (2013) 65–76.

[7] A. Mellit, A. Massi Pavan, E. Ogliari, S. Leva, V. Lughi, Advanced methods for photovoltaic output power forecasting: A review, Appl. Sci. 10 (2) (2020) 487.

[8] Y. Chu, M. Li, C.F. Coimbra, D. Feng, H. Wang, Intra-hour irradiance forecasting techniques for solar power integration: A review, Iscience 24 (10) (2021) 103136.

[9] Y. Dodge, The Concise Encyclopedia of Statistics, Springer Science & Business Media, 2008.

[10] H. Sheng, J. Xiao, Y. Cheng, Q. Ni, S. Wang, Short-term solar power forecasting based on weighted Gaussian process regression, IEEE Trans. Ind. Electron. 65 (1) (2017) 300–308.

[11] Y. Ma, Q. Lv, R. Zhang, Y. Zhang, H. Zhu, W. Yin, Short-term photovoltaic power forecasting method based on irradiance correction and error forecasting, Energy Rep. 7 (2021) 5495–5509.

[12] F.P. Kreuwel, W. Knap, M. Schmeits, J.V.-G. de Arellano, C.C. van Heerwaarden, Forecasting day-ahead 1-minute irradiance variability from numerical weather predictions, Sol. Energy 258 (2023) 57–71.

[13] X. Meng, F. Gao, T. Xu, K. Zhou, W. Li, Q. Wu, Inverter-data-driven second-level power forecasting for photovoltaic power plant, IEEE Trans. Ind. Electron. 68 (8) (2020) 7034–7044.

[14] A. Mellit, A.M. Pavan, V. Lughi, Deep learning neural networks for short-term photovoltaic power forecasting, Renew. Energy 172 (2021) 276–288.

[15] P. Li, K. Zhou, X. Lu, S. Yang, A hybrid deep learning model for short-term PV power forecasting, Appl. Energy 259 (2020) 114216.

[16] Q.-T. Phan, Y.-K. Wu, Q.-D. Phan, An approach using transformer-based model for short-term PV generation forecasting, in: 2022 8th International Conference on Applied System Innovation, (ICASI), IEEE, 2022, pp. 17–20.

[17] P.M. Garniwa, R.A. Rajagukguk, R. Kamil, H. Lee, Intraday forecast of global horizontal irradiance using optical flow method and long short-term memory model, Sol. Energy 252 (2023) 234–251.

[18] E. Pérez, J. Pérez, J. Segarra-Tamarit, H. Beltran, A deep learning model for intra-day forecasting of solar irradiance using satellite-based estimations in the vicinity of a PV power plant, Sol. Energy 218 (2021) 652–660.

[19] F. Wang, X. Lu, S. Mei, Y. Su, Z. Zhen, Z. Zou, X. Zhang, R. Yin, N. Duić, M. Shafie-khah, J.P. Catalão, A satellite image data based ultra-short-term solar PV power forecasting method considering cloud information from neighboring plant, Energy 238 (2022) 121946.

[20] L. Cheng, H. Zang, Z. Wei, T. Ding, R. Xu, G. Sun, Short-term solar power prediction learning directly from satellite Images With Regions of interest, IEEE Trans. Sustain. Energy 13 (1) (2022) 629–639, http://dx.doi.org/10.1109/tste.2021.3123476.

[21] J. Qin, H. Jiang, N. Lu, L. Yao, C. Zhou, Enhancing solar PV output forecast by integrating ground and satellite observations with deep learning, Renew. Sustain. Energy Rev. 167 (2022) 112680.

[22] T. Yao, J. Wang, H. Wu, P. Zhang, S. Li, K. Xu, X. Liu, X. Chi, Intra-hour photovoltaic generation forecasting based on multi-source data and deep learning methods, IEEE Trans. Sustain. Energy 13 (1) (2022) 607–618, http://dx.doi.org/10.1109/tste.2021.3123337.

[23] F.J. Rodríguez-Benítez, M. López-Cuesta, C. Arbizu-Barrena, M.M. Fernández-León, M.Á. Pamos-Ureña, J. Tovar-Pescador, F.J. Santos-Alamillos, D. Pozo-Vázquez, Assessment of new solar radiation nowcasting methods based on sky-camera and satellite imagery, Appl. Energy 292 (2021) 116838.

[24] H. Wen, Y. Du, X. Chen, E. Lim, H. Wen, L. Jiang, W. Xiang, Deep learning based multistep solar forecasting for PV ramp-rate control using sky images, IEEE Trans. Ind. Inform. 17 (2) (2021) 1397–1406, http://dx.doi.org/10.1109/tii.2020.2987916.

[25] T.-P. Chu, J.-H. Guo, Y.-G. Leu, L.-F. Chou, Estimation of solar irradiance and solar power based on all-sky images, Sol. Energy 249 (2023) 495–506.

[26] C. Feng, J. Zhang, W. Zhang, B.-M. Hodge, Convolutional neural networks for intra-hour solar forecasting based on sky image sequences, Appl. Energy 310 (2022) 118438.

[27] Z. Zhen, X. Zhang, S. Mei, X. Chang, H. Chai, R. Yin, F. Wang, Ultra-short-term irradiance forecasting model based on ground-based cloud image and deep learning algorithm, IET Renew. Power Gener. 16 (12) (2022) 2604–2616.

[28] Y. Fu, H. Chai, Z. Zhen, F. Wang, X. Xu, K. Li, M. Shafie-Khah, P. Dehghanian, J.P. Catalão, Sky image prediction model based on convolutional auto-encoder for minutely solar PV power forecasting, IEEE Trans. Ind. Appl. 57 (4) (2021) 3272–3281.

[29] Y. Karout, S. Thil, J. Eynard, E. Guillot, S. Grieu, Hybrid intrahour DNI forecast model based on DNI measurements and sky-imaging data, Sol. Energy 249 (2023) 541–558.

[30] V. Le Guen, N. Thome, A deep physical model for solar irradiance forecasting with fisheye images, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2020, pp. 630–631.

[31] H. Gao, M. Liu, Short-term solar irradiance prediction from sky images with a clear sky model, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2022, pp. 2475–2483.

[32] J. Liu, H. Zang, L. Cheng, T. Ding, Z. Wei, G. Sun, A transformer-based multimodal-learning framework using sky images for ultra-short-term solar irradiance forecasting, Appl. Energy 342 (2023) 121160.

[33] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An image is worth 16x16 words: Transformers for image recognition at scale, CoRR (2020) arXiv:2010.11929, arXiv:2010.11929, URL https://arxiv.org/abs/2010.11929.

[34] M. Raghu, T. Unterthiner, S. Kornblith, C. Zhang, A. Dosovitskiy, Do vision transformers see like convolutional neural networks? in: Adv. Neur. in., Vol. 34, 2021, pp. 12116–12128.

[35] T. Xiao, M. Singh, E. Mintun, T. Darrell, P. Dollár, R. Girshick, Early convolutions help transformers see better, in: Adv. Neur. in., Vol. 34, 2021, pp. 30392–30400.

[36] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, 2018, arXiv preprint arXiv:1810.04805.

[37] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems, Vol. 30, 2017.

[38] J.L. Ba, J.R. Kiros, G.E. Hinton, Layer normalization, 2016, arXiv preprint arXiv:1607.06450.

[39] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780.

[40] K. Cho, B. Van Merriënboer, D. Bahdanau, Y. Bengio, On the properties of neural machine translation: Encoder-decoder approaches, 2014, arXiv preprint arXiv:1409.1259.

[41] S. Xu, Gatton Sky Image & PV Generation 2020, The University of Queensland, 2023, http://dx.doi.org/10.48610/bd7e108.

[42] The Bureau of Meteorology, URL http://www.bom.gov.au/.

[43] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014, arXiv preprint arXiv:1412.6980.

[44] P. Ineichen, R. Perez, A new airmass independent formulation for the linke turbidity coefficient, Sol. Energy 73 (3) (2002) 151–157, http://dx.doi.org/10.1016/S0038-092X(02)00045-2, URL https://www.sciencedirect.com/science/article/pii/S0038092X02000452.

[45] K. Simonyan, A. Vedaldi, A. Zisserman, Deep inside convolutional networks: Visualising image classification models and saliency maps, 2013, arXiv preprint arXiv:1312.6034.