RESEARCH ARTICLE

Ultra-Short Term Photovoltaic Generation Forecasting Based on Data Decomposition and Customized Hybrid Model Architecture

JOOSEUNG LEE[ID]**, JIMYUNG KANG**[ID]**, SOONWOO LEE**[ID]**, AND HUI-MYOUNG OH**[ID]
Power Grid Research Division, Korea Electrotechnology Research Institute, Ansan 15588, South Korea
Corresponding author: Jooseung Lee (jseunglee@keri.re.kr)

ABSTRACT As photovoltaic (PV) systems have been successfully adopted worldwide, accurate power generation forecasting becomes increasingly essential to stable power grid operation and smart grid applications to cope with the variability of PV systems. Several data-driven models have recently been proposed for the more accurate prediction of PV power generation and have shown good performance. In particular, hybrid models that combine the characteristics of single-structure deep learning-based models have achieved better accuracies. To this end, a novel ultra-short term PV power generation forecasting model with a hybrid structure is proposed for instantaneous response to PV fluctuations. For higher forecasting accuracy, the proposed model decomposes the input feature data into trend and residual components and employs customized sub-models such as the linear, Transformer, and long short-term memory (LSTM). Furthermore, the proposed model is trained with data from the self-built PV site to implement a model suitable to real-world applications. Finally, the experimental results demonstrate that the proposed model has the best forecasting performance compared to conventional and state-of-the-art deep learning-based forecasting models with reasonable computational complexity.

INDEX TERMS Deep learning, renewable energy, photovoltaic, solar, microgrid, ultra-short term forecasting, energy AI.

I. INTRODUCTION

Renewable energy generation technologies have rapidly advanced as environmental protection policies such as carbon neutrality become increasingly emphasized worldwide [1]. Accordingly, renewable energy power plants have replaced traditional ones, such as thermal power plants, that emit carbon and pollute the environment [2]. The importance of distributed energy resources (DERs) is also emphasized in South Korea, and DERs are expected to account for 25–40% of the total power generation by 2040 [3]. One of the most popular DERs today is the photovoltaic (PV) system, owing to its relatively easy installation and high power generation efficiency. However, there are several challenges to stable

The associate editor coordinating the review of this manuscript and approving it for publication was Ahmed F. Zobaa[ID].

power grid operation with a PV system because it is sensitive and highly volatile to weather conditions and electrical faults [4]. For reliable power grid operation, accurate forecasting of future PV power generation is essential.

PV power generation forecasting can be categorized into several types based on prediction time [5], [6], [7]. Among these, ultra-short term forecasting, which predicts power generation for the next hour or several hours, is important for responding to the variability of PV power generation in real time. Whereas traditional power plants capable of controlling output power operate based on load data, future power plants, including DERs, should consider irregular power generation. In other words, power plants with many DERs suffer from overvoltage problems and commonly known power shortage issues [8]. In this regard, most countries, including South Korea, use 15-minute load data profiles for power grid

operations [9], [10]. Previous studies have also demonstrated that at least a 15-minute power generation forecast is required for microgrid backup systems [11].

Additionally, ultra-short term PV forecasting is crucial for smart grid applications that require real-time control. For example, when using a PV-linked energy storage system (ESS), ultra-short term PV forecasting is employed to control the ESS charging in response to PV volatility [12]. Furthermore, for a distributed conservation voltage reduction (CVR) [13] with continuous voltage regulation [14], [15], power generation data predicted in short time units can be used as key information for energy optimization. These applications make ultra-short term PV generation forecasting an essential area in PV systems research.

However, as explained earlier, PV power generation varies in different patterns depending on environmental conditions, making it difficult to forecast future power generation accurately [16]. Power generation can fluctuate almost randomly within minutes owing to the influence of irregular clouds [17]. Many studies have, therefore, worked on solving this problem through statistical and machine learning approaches using measured data. Among these statistical methods, Sheng et al. proposed weighted Gaussian process regression (WGPR) to predict PV generation after 5 minutes and confirmed performance gains compared to support vector machine (SVM), artificial neural network (ANN), and normal GPR method [18]. Additionally, fuzzy logic was proposed in [19] to forecast future 15-minute power generation by finding statistical relationships between feature data.

To improve forecasting performance, proven deep learning models have been adopted for ultra-short term forecasting. For example, the authors of [20] presented a self-developed optimization method for the ANN to predict power generation for the next 1 hour. Subsequently, prediction models employing long short-term memory (LSTM) [21] and gated recurrent unit (GRU) structures of the recurrent neural network (RNN) family, which have good performance in time series prediction, have been proposed and exhibited high prediction accuracies [17], [22]. Furthermore, to overcome the limitations of traditional RNN models that use compressed historical information, the attention-based sequence-to-sequence (Seq2seq) model, which considers all the associations between data at each time point, was adopted in [23]. Recently, the Transformer model [24], which maximizes the benefits of the attention approach, has shown good performance in language models and was also used in PV generation forecasting [25].

Many hybrid models combining proven deep learning models have also been proposed to enhance forecasting accuracy. In order to consider both the spatial characteristics between features and the temporal characteristics, hybrid models that combine RNN and convolutional neural network (CNN) models were proposed in [26], [27], and [28], leading to better performance than conventional single-structure models. Recently, hybrid models that decompose input data and apply individual models have been proposed to improve

TABLE 1. Key nomenclature for the symbols and abbreviations.

Key nomenclature for the symbols

Symbol	Description
ρ	Pearson correlation coefficient.
$Cov(x, y)$	Covariance of x and y.
$E(x)$	Expected value of x.
σ_x	Standard deviation of x.
X	Input matrix of a layer.
Y, y	Output matrix and a single output of a layer.
W, W, U	Weight matrix and a single weight of a layer.
B, b	Bias matrix and a single bias of a layer.
Q, K, V	Query, key, and value matrix for the Transformer.
A_h	Attention matrix of a head for the Transformer.

Key abbreviations

Abbrev.	Description
PV	Photovoltaic
DER	Distributed energy resource
ANN	Artificial neural network
RNN	Recurrent neural network
LSTM	Long short-term memory
Seq2seq	Sequence to sequence
CNN	Convolutional neural network
SOTA	State-of-the-art
FEDformer	Frequency enhanced decomposed Transformer
SCINet	Sample convolution and interaction network
PE	Positional encoding
MHA	Multi-head attention
FFNN	Feedforward neural network
MSE	Mean squared error
MAE	Mean absolute error
EMAPE	Effective mean absolute percentage error

prediction accuracy further. For example, Yan et al. achieved performance gains compared to conventional deep learning algorithms by separating data through fast Fourier transform (FFT) and applying individual CNNs to low- and high-frequency components [29]. Similarly, a data decomposition method using a wavelet transform (WT) was employed in [30] and [31], and it led to good forecasting performance. As the performance contribution of the data decomposition approach is continuously proven in general multi-point time series forecasting applications, most high-performance state-of-the-art (SOTA) models, such as the frequency enhanced decomposed Transformer (FEDformer) [32], sample convolution and interaction network (SCINet) [33], and N/Dlinear [34], have adopted this approach. For these SOTA models, methods such as mixed transform, auto correlation-based attention, tree-structured sampling, iterative convolution, and multi-wise linear processing are also applied to capture short- to mid-term input feature data patterns and predict future targets accurately.

In this brief, a novel hybrid model architecture is proposed for intra-hour ultra-short term PV power generation forecasting. The proposed model decomposes PV-related feature data in simple but appropriate ways and employs customized
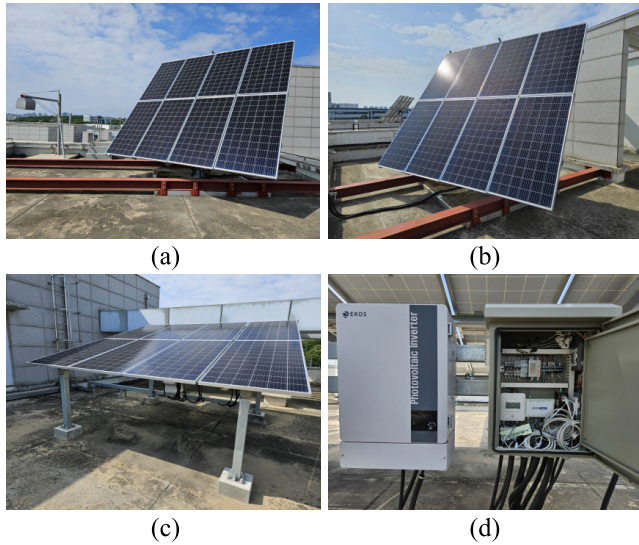
FIGURE 1. PV site for study: (a) biaxial solar tracking panel, (b) uniaxial solar tracking panel, (c) fixed panel, (d) inverter, and sensor network.

TABLE 2. Input features for proposed model.

Feature	Description	ρ
PV power generation	Target. Amount of power generated by a panel per unit time.	1
Solar irradiance on tilted panel	Solar irradiance reaching the tilted slope of a panel.	0.98
Global horizontal irradiance	Total solar radiation reaching the horizontal surface of the ground.	0.92
Temperature inside a panel	Internal air temperature of a panel.	0.61
Temperature outside a panel	External air temperature of a panel.	0.31
Temperature	Ambient temperature.	0.27
Humidity	Ambient humidity.	0.56
Dust concentration	Ambient dust concentration. (PM 2.5)	0.05
Wind speed	Ambient wind speed.	0.33
Wind direction	Ambient wind direction.	-
Time	Month, hour, and minute.	-

sub-models. The key points of this study can be summarized as follows.

1. An average pooling is applied to decompose the input feature data into trend and residual components.
2. The linear sub-model is employed for the trend component, which is characterized by relatively slight variation, and the Transformer and LSTM-based sub-models are used to process the residual components, which are irregular and highly fluctuating.
3. The proposed model is trained and validated using feature data collected from the self-built PV site for a practical evaluation in real-world applications.
4. Performance gains of the proposed model relative to proven RNN-based single models, a CNN-RNN-based hybrid model, and SOTA hybrid models are presented to prove novelty in forecasting performance.

The rest of this paper is organized as follows. First, the details of data used to implement forecasting models are described in Section II. Section III provides the detailed structure of the proposed model. The experiments' setup and results to evaluate the performance of the proposed model are described in Section IV. Finally, the conclusion of this study is presented in Section V.

II. DATA COLLECTION

This section presents an overview of the data for the study. Instead of relying on benchmark datasets, a PV site is built to implement an ultra-short term PV forecasting model. This makes it possible to develop a model that is more suitable for real-world applications and to evaluate the performance of the proposed model more practically.

As shown in Fig. 1, the PV site, located in Ansan, South Korea, consists of several types of solar panels, inverters, and a sensor network. To account for the characteristics of different PV panel types, data from three different types of 3 kW solar panels are obtained: biaxial solar tracking, uniaxial solar tracking, and fixed panels. Electrical and panel-related environmental data are acquired from the panels and connected inverters, and other site-wide meteorological data are measured through additional sensors. All data obtained from the devices are transmitted to remote terminal units (RTUs), and the RTUs send the data to the database server.

The detailed feature data used to implement the proposed model are summarized in Table 2. Because the proposed model targets ultra-short term forecasting within an hour, it uses only local measurement data, not weather forecast data that provide information after an hour. In order to identify the association between each feature and the PV power generation numerically, the correlation coefficients are calculated and presented in the table. The correlation coefficient ρ between data x and y is calculated as follows.

$$\rho = \frac{\text{Cov}(x, y)}{\sigma_x \sigma_y} \quad (-1 \leq \rho \leq 1) \quad (1)$$

$$\text{Cov}(x, y) = \text{E}[(x - \text{E}(x))(y - \text{E}(y))] \quad (2)$$

where $\text{Cov}(x,y)$ is the covariance of x and y; $\text{E}(x)$ and $\text{E}(y)$ are the means of x and y, respectively; and σ_x and σ_y are the standard deviations of x and y, respectively. The higher the correlation, the larger the absolute value of the coefficient ρ.

Table 2 shows that solar irradiance data correlate most with PV power generation. Accordingly, they are used as the primary input for the proposed model, along with the historical PV power generation patterns. The other data in the table are also employed as supplementary inputs to the model to predict detailed variations in the target. Furthermore, since solar irradiance and PV power generation data have an annual cycle and daily patterns, each data measurement's month, hour, and minute values are also used as timestamp data.
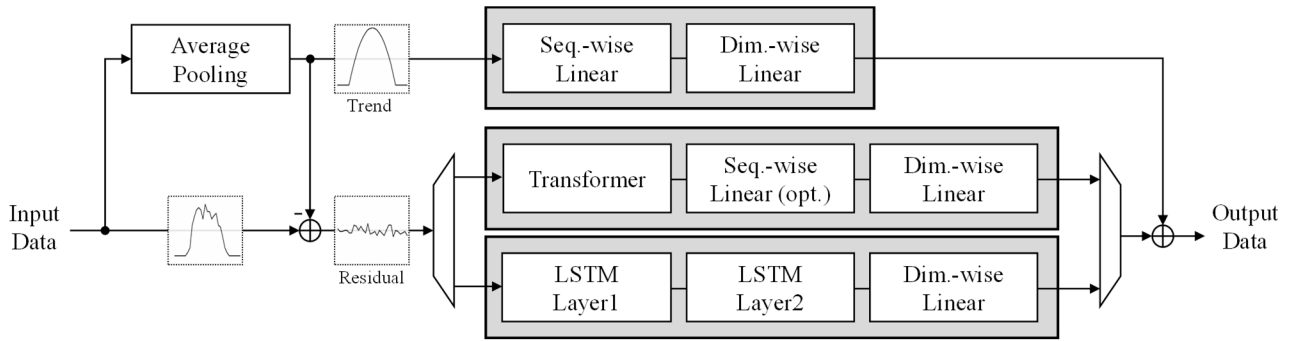
FIGURE 2. Overall architecture of proposed PV power generation forecasting model.
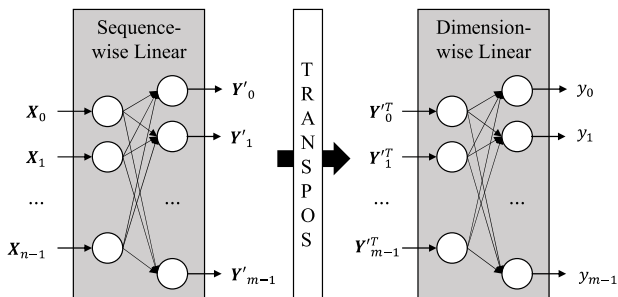

FIGURE 3. Linear sub-model for trend component.

In conclusion, the proposed model utilizes the historical patterns of the feature data listed in Table 2 as input data to forecast future PV power generation patterns.

III. MODEL ARCHITECTURE

This section describes the overall architecture and details of the proposed model. As summarized in Fig. 2, the proposed model consists of the linear sub-model, and the Transformer and LSTM-based sub-models. In general, the PV-related feature data have approximately linear trend curves in short time intervals and irregular residual variations that depend on the positions of the sun and clouds. In this respect, each feature data can be processed separately to achieve better forecasting accuracy. To this end, the proposed model decomposes the original input data into trend and residual components by average pooling kernel and employs customized sub-models for each data component. The final PV power generation output is calculated as the weighted sum of the trend and residual components through the linear sub-layers included in the last stages of the sub-models and the addition operator.

A. SUB-MODEL FOR TREND

As previously summarized, the linear sub-model produces the trend component of the final output, and the detailed structure is depicted in Fig. 3. It is a multivariate to univariate ANN model that produces 1-dimensional output data with d_f-dimensional input data. It is also a multi-horizon forecasting model that generates m future output points with n

historical input points. In other words, the sub-model consists of sequence- and dimension-wise linear sub-layers to obtain the final output. This approach of processing the trend component with an ANN model is common in SOTA models and works successfully for the regression of the trend component [33], [34].

First, the d_f-dimensional input feature data vector X with sequence length n is processed in the first linear sub-layer in a sequential direction to produce d_f-dimensional m-horizon intermediate output Y'. The sub-layer can be configured with a single linear layer for all features or individual linear layers for each feature, and the final structure should be determined considering the performance advantages and computational complexity. Furthermore, the sub-layer may have multiple hidden layers and dropout layers to achieve better results. Finally, the operation of the first sub-layer is mathematically expressed as follows.

$$Y'(k) = W(k)X(k) + B(k) \qquad (3)$$

where $X(k)$ and $Y'(k)$ are the input and output matrices of the k-th linear layer, respectively, and W and B are the weight and bias matrices of the layer.

Finally, to produce the trend component output for future PV power generation, the transposed intermediate output of the first sub-layer Y' is used as an input vector for the second sub-layer. For the final univariate output, the second sub-layer processes the input data in the dimensional direction. The second sub-layer may also have multiple hidden layers and dropout layers, and the basic calculation is the same as for the first sub-layer, expressed in (3). In summary, the final output y of the first sub-model is the trend component of the m-horizon future PV power generation data.

B. SUB-MODEL FOR RESIDUAL

As depicted in Fig. 2, the model for the residual component, after the trend part is removed from the original input data, consists of the Transformer and LSTM-based sub-models. These two types of sub-models have complementary advantages in terms of forecasting accuracy and computational complexity and can be activated selectively depending on the system requirements. Although SOTA models for time
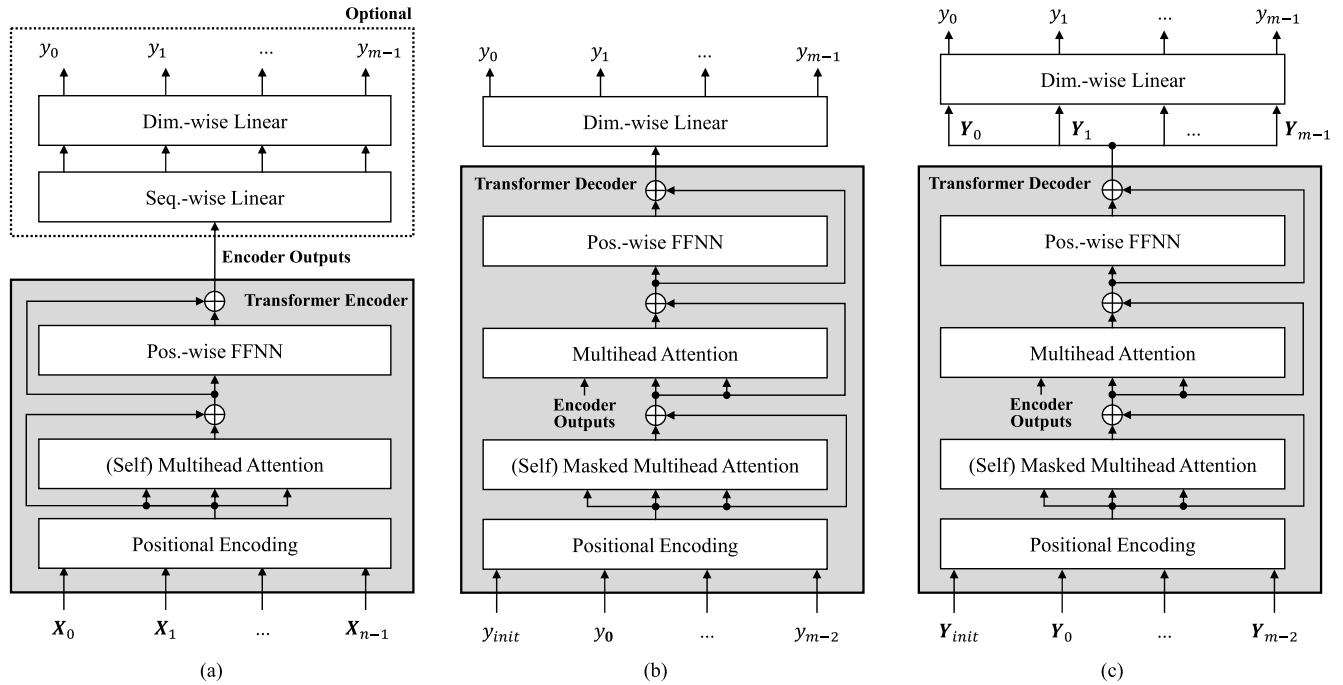
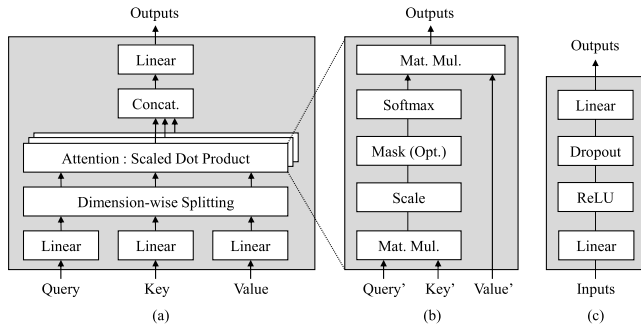**FIGURE 4.** Transformer-based sub-model for residual component: (a) Encoder and (b, c) decoder structures.

FIGURE 5. Components of transformer: (a) MHA, (b) scaled dot product, and (c) position-wise FFNN.

conventional Transformer model are available for the forecasting application, which will be presented further in the paper.

As shown in Fig. 4, the Transformer encoder and decoder consist of positional encoding, multi-head attention (MHA), and position-wise feedforward neural network (FFNN) components. First, positional encoding is applied to retain the temporal order information of the input data sequence, like the RNN. The positional information for the data at time t is computed as follows [24].

$$\text{PE}_{(t, i_{\text{even}})} = \sin(t/10000^{2i/d_{\text{input}}}) \quad (4)$$

$$\text{PE}_{(t, i_{\text{odd}})} = \cos(t/10000^{2i/d_{\text{input}}}) \quad (5)$$

where i is the position in dimension, and d_{input} is the dimension of the input vector. Through the addition of the calculated positional vector to the original input vector, the input data, including the temporal-order information, are prepared for processing.

Subsequently, the Transformer computes the MHA to determine the temporal relationships of the data sequence. As shown in Fig. 4, the Transformer encoder calculates only its self MHA, which determines the temporal dependencies of the input data sequence. By contrast, the decoder also evaluates the MHA that identifies the relationship between the encoder output and decoder input. The detailed processing flow for the MHA is shown in Fig. 5. First, raw input data are projected through multiple linear layers to derive the query, key, and value required to calculate the attention. Subsequently, the query, key, and value are divided into several heads in the dimensional direction, and a scaled dot

series forecasting have focused on finding additional periodicity (seasonality) in the residual region, it is observed that the residual part of the PV data usually fluctuates according to the random distribution of clouds; therefore, the sub-model structure to capture the attention between all the residual points and the compressed context vector is finally adopted.

The detailed structure of the Transformer-based sub-model is shown in Fig. 4. The Transformer model focuses on computing attention values, which represent the associations between data at each time point and are further processed in the fully connected ANN layer. As described earlier, this approach makes it possible to model residual components with fluctuations at each time point. The conventional Transformer model is an encoder-decoder model in which the decoder utilizes compressed information from the encoder to produce the final output. Several variations of the

FIGURE 6. LSTM-based sub-model for residual component.

FIGURE 7. Overall flow of experiment.

product attention vector between each point of the sequence is computed for each head. As shown in Fig. 5 (b), the scaled dot product attention vector is calculated through matrix multiplication, scaling, and softmax and summarized by the following formula [24].

$$\text{Attention}\,(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{softmax}(\boldsymbol{Q}\boldsymbol{K}^{\text{T}}/\sqrt{d_k})\boldsymbol{V} \qquad (6)$$

where $\boldsymbol{Q}, \boldsymbol{K}$, and $\boldsymbol{V}$ refer to the projected query, key, and value, respectively, and d_k is the dimension of the key. Finally, the attention vector is obtained through the linear layer for the concatenated attention values of all the heads as follows [24].

$$\text{MHA}\,(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{Concat}\,\big(\boldsymbol{A}_{h_0}, \boldsymbol{A}_{h_1}, \ldots, \boldsymbol{A}_{h_k}\big) \times \boldsymbol{W}_{\text{o}} \quad (7)$$

where $\boldsymbol{A}_{h_i}$ is the attention matrix of the i-th head, and $\boldsymbol{W}_{\text{o}}$ is the weight matrix of the last linear layer.

Finally, the poison-wise FFNN layer produces the final output data. The FFNN layer comprises two linear layers, a rectified linear unit (ReLU) activation function, and a dropout layer, as shown in Fig. 5 (c). The corresponding layer output is summarized as follows [24].

$$\text{FFNN}\,(\boldsymbol{X}) = \max\,(0, \boldsymbol{X}\boldsymbol{W}_1 + \boldsymbol{B}_1)\,\boldsymbol{W}_2 + \boldsymbol{B}_2 \qquad (8)$$

where $\boldsymbol{X}$ is the input vector of the FFNN layer, and $\boldsymbol{W}_i$ and $\boldsymbol{B}_i$ are the weight and bias vectors of the i-th linear layer.

To summarize, the Transformer model has an encoder and a decoder composed of these sub-functions. The added and normalized output vector from the previous sub-function layer is used as an input to improve the training performance of every sub-function. Finally, the Transformer model can be configured with sequentially connected multiple encoders and decoders depending on the data size to obtain better results.

As mentioned, the conventional Transformer model should be modified for PV power generation forecasting. The encoder-only model shown in Fig. 4 (a) can be used for the residual part with the optional layer. The model first applies the Transformer encoding to a multivariate input data vector $\boldsymbol{X}$ of length n and produces an intermediate output vector with the same shape as the input vector. Then, the intermediate output vector is processed through sequence- and dimension-wise linear layers to produce the final output with the optional layer. In other words, it is a forecasting model utilizing only the temporal relationship of historical
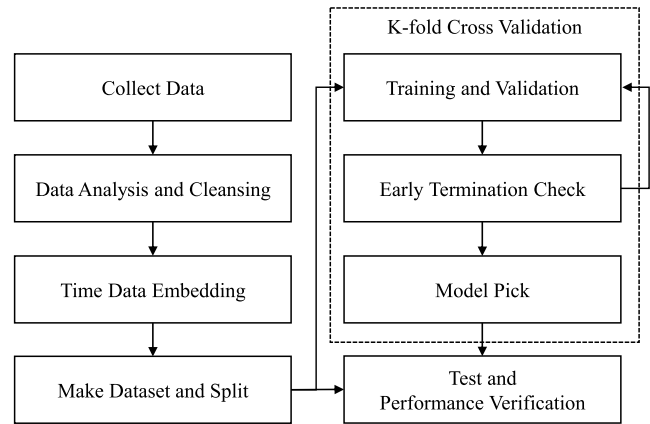
feature data and simultaneously producing target data for m future points.

Meanwhile, two types of encoder-decoder models with Fig. 4 (b) and (c) structures can be implemented. Unlike the encoder-only model, these models consider the temporal associations, including future target data to be produced in the decoder, and thus, better performance is expected. Whereas the encoder-only model computes the future output simultaneously, these encoder-decoder models generate the future data iteratively through the decoder. In other words, m-times decoding is required to produce the PV power generation output for m future points, expressed as dynamic decoding in the following.

For the encoder-decoder models, a Transformer encoder excluding the optional layer is used, and two types of decoders can be employed. If only the univariate data, i.e., the historical PV power generation data, are used as the decoder input, the decoder is configured as Fig. 4 (b). This decoder considers only the attention for the final target y with the encoder output. Accordingly, it obtains the i-th output y_i through dynamic decoding and uses an intermediate decoder output as an input for the next decoding. Since the decoder has a multi-dimensional output, a dimension-wise linear layer is required for every decoding step to obtain the univariate output y. Meanwhile, the most recent historical PV power generation value is used as an initial input of the decoder. Thus, this encoder-decoder model requires one encoding, m-times dynamic decoding, and m-times dimension-wise linear layer computations to generate m future points.

Next, the decoder structure presented in Fig. 4 (c), which directly uses the Transformer decoder output vector for the next decoding, can also be employed. Because a multivariate vector $\boldsymbol{Y}$ is used for dynamic decoding, a dimension-wise linear layer is required only after the last decoding to obtain the final univariate output y. Thus, this structure requires one encoding step, m-times decoding, and one dimension-wise linear layer computation to obtain m points of the future target. This structure allows for better performance because the information generated by the Transformer decoder is not compressed and is fully utilized for dynamic decoding.
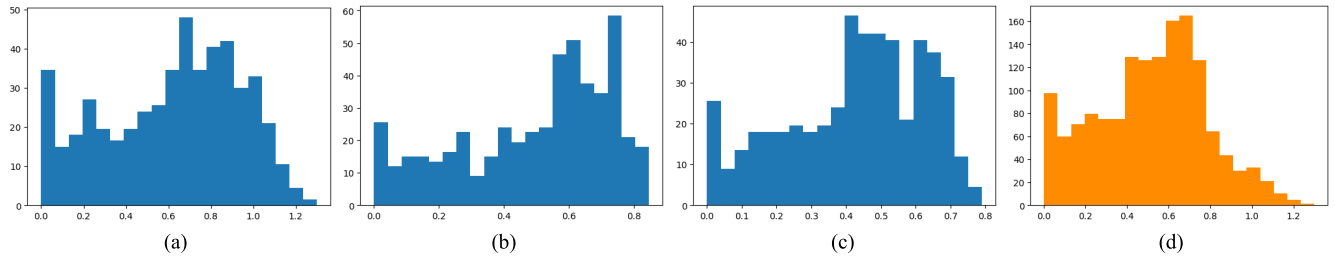
FIGURE 8. Histograms of average hourly PV power generation per day (kW): (a) Biaxial, (b) uniaxial solar tracking, (c) fixed panels and (d) total.

Depending on the system requirements, the LSTM-based sub-model can be used as an alternative for the residual component processing. The corresponding structure, which has two LSTM layers and one linear layer, is shown in Fig. 6. The multivariate residual component X is used as an input for the first LSTM layer, and the output hidden state vector of the first layer is further processed in the second LSTM layer. Each LSTM layer may have multiple cells, and in general, the number of cells should be larger than the dimension of the input vector for better performance. For a multiple-cell structure, the final residual component output is obtained through the dimension-wise linear layer.

The basic structure of a single LSTM cell is shown in Fig. 6 (b). It consists of multiple gates that operate according to the following formulas at time t [21].

$$\text{Forget gate: } f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \qquad (9)$$

$$\text{Input gate-1: } i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \qquad (10)$$

$$\text{Input gate-2: } c'_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \qquad (11)$$

$$\text{Output gate-1: } o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \qquad (12)$$

$$\text{Output gate-2: } h_t = o_t \times \tanh(c_t) \qquad (13)$$

$$\text{Cell state: } c_t = f_t \times c_{t-1} + i_t \times c'_t \qquad (14)$$

where σ is a sigmoid operator, x_t is an input, c_t is a cell state output, and h_t is a hidden state output. For the trainable model parameters, W and U are the weight values for x_t and h_{t-1}, respectively, and b is a bias term. For a multiple-cell LSTM model, each cell has individual parameters and is trained independently.

IV. EXPERIMENTAL RESULTS

This section covers the experimental setup, results, and analysis to evaluate the performance of the proposed model. First, the experimental setup, including the dataset and details of the models, is described. Then, numerical and visual results of the models in the proposed and comparison groups are presented, followed by detailed discussions.

A. EXPERIMENT SETUP

The overall experimental process is illustrated in Fig. 7. First, 18 months of data from April 2022 to September 2023 were utilized. The data were collected from three types of 3 kW solar panels in the self-built PV site. All the local measurement data presented in Table 2 were utilized as historical input features for the models, and the data resolution is 5 minutes to account for forecasting accuracy and computational efficiency.

The data for the experiment were measured under four distinct seasons of climatic conditions in South Korea. To numerically verify whether the data cover various weather conditions, histograms of the average hourly PV power generation per day in kilowatts are presented with 20 bins in Fig. 8. In the figure, the clearer the weather, the higher the average daily power generation, and the generation of the biaxial solar tracking panel is relatively greater than that of the other types. The overall distribution shows that days with average power generation in the middle to upper range are relatively dominant, but the data also include many weather conditions with average power generation in the low range.

For data cleansing, data acquired on days when the PV system was not normally operating are removed. Then, outliers are eliminated by considering the measurement range of the devices, and the corresponding abnormal data points and missing points are interpolated. Subsequently, time data embedding is applied to the timestamp data, allowing the timestamps to express periodicity and continuity through the Euler formula with trigonometric functions.

After data preprocessing, the datasets are finally created by the sliding window method with a stride size of one. Then, the dataset is normalized for effective training and is allocated 60%, 20%, and 20% for training, validation, and testing. K-fold cross-validation is used to evaluate performance across all seasons and weather conditions in the data. In addition, all the models are trained with different random seeds for each fold, and the average is obtained.

For the training process, the mean squared error (MSE) is used as the criterion. A validation loss is calculated for the validation dataset at each learning epoch; if the loss does not decrease for the predefined patience counts, the training is terminated early. Finally, the trained models are evaluated using the test dataset based on several metrics. Details on the evaluation criteria are provided further in the paper.

The detailed structures of the models in the proposed and comparison groups are described in Table 3. For the proposed group, the performances of the four configurations are observed. First, the sub-model for the trend component is classified into the structure using a single linear layer for all features and the structure with individual linear

TABLE 3. Detailed structures of models in proposed and comparison groups.

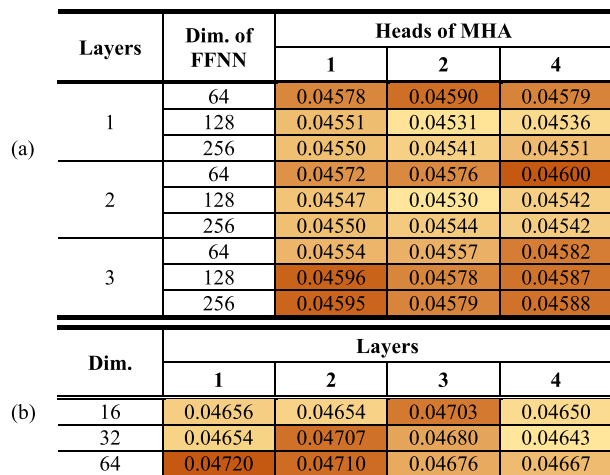| Model	Structure
Comparison Group	
LSTM	LSTM (16 dims., 2 layers) – Dim.-wise Linear (1 layer)
Seq2seq (with attention)	LSTM (16 dims., 2 layers) encoder – LSTM (16 dims., 1 layer) decoder with dot-product attention – Dim.-wise Linear (1 layer)
TF	Transformer encoder – decoder (multi dim. input as Fig. 4 (c)) – Dim.-wise Linear (1 layer) (Transformer encoder, decoder: 1 layer, 2 heads, 128 dims. for position-wise FFNN)
CNN_LSTM	Conv1D (kernel size 3, stride 1) – Conv1D (kernel size 3, stride 1) – MaxPool (kernel size 2, stride 2) – ReLU – LSTM (16 dims., 2 layers) – Dim.-wise Linear (1 layer)
FF	FEDformer encoder – decoder – Dim.-wise Linear (1 layer) (FEDformer encoder, decoder: Wavelet mode, 1 layer, 2 heads, 128 dims. for position-wise FFNN)
SCINet	Avg. Pool. – [Trend : Seq.-wise Linear (1 layer for all features)], [Residual : SCINet (1 stack, 2 levels, 1 group)] – Merge – Dim.-wise Linear (1 layer)
Dlinear	Avg. Pool. – [Trend : Seq.-wise Linear (1 layer for each feature)], [Residual : Seq.-wise Linear (1 layer for each feature)] – Merge – Dim.-wise Linear (1 layer)
Proposed Group	
DC_LSTM_I	Avg. Pool. – [Trend : Seq.-wise Linear (1 layer for each feature) – Dim.-wise Linear (1 layer)], [Residual : LSTM (16 dims., 2 layers) – Dim.-wise Linear (1 layer)] – Merge
DC_LSTM_S	Avg. Pool. – [Trend : Seq.-wise Linear (1 layer for all features) – Dim.-wise Linear (1 layer)], [Residual : LSTM (16 dims., 2 layers) – Dim.-wise Linear (1 layer)] – Merge
DC_TF_I	Avg. Pool. – [Trend : Seq.-wise Linear (1 layer for each feature) – Dim.-wise Linear (1 layer)], [Residual : Transformer encoder – decoder (multiple dim. input as Fig. 4 (c)) – Dim.-wise Linear (1 layer)] – Merge (Transformer encoder, decoder: 1 layer, 2 heads, 128 dims. for position-wise FFNN)
DC_TF_S	Avg. Pool. – [Trend : Seq.-wise Linear (1 layer for all features) – Dim.-wise Linear (1 layer)], [Residual : Transformer encoder – decoder (multiple dim. input as Fig. 4 (c)) – Dim.-wise Linear (1 layer)] – Merge (Transformer encoder, decoder: 1 layer, 2 heads, 128 dims. for position-wise FFNN)

(a)

Layers	Dim. of FFNN	Heads of MHA		
		1	2	4
1	64	0.04578	0.04590	0.04579
	128	0.04551	0.04531	0.04536
	256	0.04550	0.04541	0.04551
2	64	0.04572	0.04576	0.04600
	128	0.04547	0.04530	0.04542
	256	0.04550	0.04544	0.04542
3	64	0.04554	0.04557	0.04582
	128	0.04596	0.04578	0.04587
	256	0.04595	0.04579	0.04588

(b)

Dim.	Layers			
	1	2	3	4
16	0.04656	0.04654	0.04703	0.04650
32	0.04654	0.04707	0.04680	0.04643
64	0.04720	0.04710	0.04676	0.04667

FIGURE 9. MSE heatmaps for sub-models in proposed model with respect to hyperparameters: (a) Transformer and (b) LSTM.

previous studies are considered. First, the LSTM, Seq2seq with attention, and Transformer (without data decomposition) models are selected for the single-structure comparative models. Additionally, the CNN_LSTM model is chosen for the CNN-RNN-based hybrid model, and the FEDformer, SCINet (data decomposition enabled), and Dlinear models are also compared as the SOTA hybrid models with data decomposition. Because the models target multi-point forecasting, a persistent model that produces the last measured value as a straightforward future prediction is not considered.

The hyperparameters of all models were configured through preliminary grid search experiments, considering the model's complexity and performance gain. In order to explain the preliminary experiments in more detail, an example of the grid search to determine the hyperparameters of the sub-models in the proposed model is presented in Fig. 9. The MSE losses of the sub-models are presented with respect to the hyperparameters; the smaller the value, the better the forecasting accuracy. For the Transformer-based sub-model, the numbers of encoder and decoder layers, the dimension of the FFNN, and the number of heads in the MHA are hyperparameters. This sub-model performs better when it is configured with one or two layers, 128 dimensions, and two heads. In this case, it is reasonable to choose the one-layer structure because the two-layer structure has twice the computational complexity, but the performance gain is only 0.02%. For the LSTM-based sub-model, the results show that the most reasonable score is achieved with the configuration of 16 cells and two layers.

layers for each feature. In the table, the former and latter cases are denoted with '_S' and '_I', respectively. Second, the sub-model for the residual component is divided into the Transformer and LSTM-based types. In the preliminary experiment, the encoder-decoder structure configured as Fig. 4 (c) outperformed the structures of Fig. 4 (a) and (b) by more than 12% and 3%, respectively, and was finally adopted for the Transformer-based proposed model.

For the comparison group, conventional single and hybrid models whose performances have already been proven in

TABLE 4. (a) Forecasting performance comparison: MSE (Normalized scale). (b) Forecasting performance comparison: MAE (Normalized scale). (c) Forecasting performance comparison: EMAPE (%).

(a)

Output Length (min.)	Input Length (min.)	Model										
		Comparison Group							Proposed Group			
		LSTM	Seq2seq	TF	CNN-LSTM	FF	SCINet	Dlinear	DC_LSTM_I	DC_LSTM_S	DC_TF_I	DC_TF_S
5	10	0.0322	0.0305	0.0374	0.0310	0.0310	-	0.0314	0.0307	0.0307	0.0304	**0.0303**
5	20	0.0314	0.0307	0.0350	0.0309	0.0324	0.0309	0.0309	0.0300	0.0300	**0.0297**	0.0297
5	30	0.0309	0.0304	0.0350	0.0295	0.0324	-	0.0306	0.0291	0.0290	**0.0287**	0.0289
10	20	0.0429	0.0426	0.0456	0.0433	0.0408	0.0412	0.0426	0.0404	0.0403	**0.0398**	0.0402
10	40	0.0425	0.0416	0.0444	0.0401	0.0402	0.0406	0.0418	0.0375	0.0375	**0.0375**	0.0375
10	60	0.0414	0.0408	0.0423	0.0391	0.0393	0.0401	0.0404	0.0368	0.0366	**0.0366**	0.0366
15	30	0.0512	0.0504	0.0534	0.0497	0.0491	-	0.0513	0.0463	0.0462	**0.0456**	0.0456
15	60	0.0493	0.0485	0.0505	0.0476	0.0465	0.0474	0.0488	0.0429	0.0435	0.0430	**0.0427**
15	90	0.0484	0.0474	0.0498	0.0471	0.0467	-	0.0476	0.0426	0.0429	0.0426	**0.0425**
20	40	0.0583	0.0575	0.0606	0.0562	0.0572	0.0555	0.0588	0.0500	0.0500	0.0493	**0.0490**
20	60	0.0571	0.0551	0.0572	0.0547	0.0537	0.0526	0.0558	0.0485	0.0487	0.0482	**0.0479**
20	80	0.0566	0.0544	0.0561	0.0544	0.0533	0.0509	0.0548	0.0483	0.0485	0.0475	**0.0471**
40	80	0.0836	0.0781	0.0796	0.0761	0.0811	0.0691	0.0799	0.0674	0.0668	0.0658	**0.0653**
40	120	0.0833	0.0777	0.0799	0.0749	0.0801	0.0676	0.0777	0.0687	0.0681	**0.0649**	0.0650
40	160	0.0831	0.0781	0.0794	0.0754	0.0797	0.0670	0.0765	0.0685	0.0676	0.0650	**0.0646**
60	120	0.1041	0.1009	0.1024	0.0980	0.1040	0.0848	0.1015	0.0881	0.0863	0.0825	**0.0817**
60	180	0.1046	0.0992	0.1028	0.0984	0.1021	0.0835	0.0993	0.0858	0.0863	0.0818	**0.0813**
60	240	0.1036	0.0981	0.1046	0.0987	0.1026	0.0842	0.0987	0.0878	0.0866	0.0812	**0.0811**
Average Loss		0.0614	0.0590	0.0620	0.0581	0.0596	0.0540	0.0594	0.0527	0.0525	0.0511	**0.0509**
Relative Loss to the Best (%)		20.44	15.83	21.67	14.00	16.92	5.96	16.50	3.51	3.11	0.32	Best

(b)

Output Length (min.)	Input Length (min.)	Model										
		Comparison Group							Proposed Group			
		LSTM	Seq2seq	TF	CNN-LSTM	FF	SCINet	Dlinear	DC_LSTM_I	DC_LSTM_S	DC_TF_I	DC_TF_S
5	10	0.0710	0.0677	0.1045	**0.0647**	0.0672	-	0.0702	0.0670	0.0672	0.0655	0.0656
5	20	0.0715	0.0672	0.1005	**0.0635**	0.0700	0.0696	0.0688	0.0655	0.0657	0.0647	0.0648
5	30	0.0703	0.0666	0.1012	**0.0617**	0.0710	-	0.0682	0.0641	0.0636	0.0627	0.0629
10	20	0.0840	0.0850	0.1044	0.0846	0.0808	0.0832	0.0869	0.0790	0.0786	**0.0784**	0.0788
10	40	0.0844	0.0829	0.0998	0.0802	0.0808	0.0827	0.0855	0.0749	0.0750	**0.0734**	0.0745
10	60	0.0828	0.0820	0.0948	0.0793	0.0794	0.0825	0.0836	0.0737	0.0733	**0.0731**	0.0731
15	30	0.0945	0.0950	0.1107	0.0921	0.0934	-	0.0993	0.0882	0.0879	0.0856	**0.0852**
15	60	0.0927	0.0926	0.1051	0.0884	0.0909	0.0920	0.0969	0.0829	0.0836	0.0820	**0.0813**
15	90	0.0916	0.0913	0.1033	0.0880	0.0951	-	0.0951	0.0832	0.0838	0.0816	**0.0815**
20	40	0.1015	0.1024	0.1174	0.0989	0.1082	0.1015	0.1106	0.0936	0.0933	0.0902	**0.0894**
20	60	0.1012	0.1006	0.1130	0.0972	0.1023	0.0994	0.1083	0.0922	0.0917	0.0896	**0.0894**
20	80	0.1010	0.0997	0.1098	0.0969	0.1044	0.0960	0.1069	0.0921	0.0918	0.0883	**0.0870**
40	80	0.1273	0.1252	0.1331	0.1211	0.1321	0.1198	0.1447	0.1189	0.1177	0.1137	**0.1121**
40	120	0.1272	0.1238	0.1316	0.1169	0.1307	0.1165	0.1413	0.1224	0.1229	**0.1116**	0.1129
40	160	0.1260	0.1227	0.1316	0.1177	0.1291	0.1171	0.1389	0.1225	0.1223	0.1123	**0.1112**
60	120	0.1431	0.1431	0.1488	0.1403	0.1517	0.1363	0.1725	0.1483	0.1458	0.1343	**0.1329**
60	180	0.1451	0.1404	0.1474	0.1378	0.1476	0.1362	0.1690	0.1454	0.1472	**0.1328**	0.1343
60	240	0.1440	0.1407	0.1479	0.1376	0.1455	0.1359	0.1677	0.1491	0.1493	**0.1353**	0.1355
Average Loss		0.1033	0.1016	0.1169	0.0982	0.1044	0.0995	0.1119	0.0979	0.0978	.0931	**0.0929**
Relative Loss to the Best (%)		11.17	9.36	25.85	5.64	12.41	7.13	20.43	5.41	5.28	0.16	Best

To evaluate the performance of the models from various perspectives, MSE, mean absolute error (MAE), and effective mean absolute percentage error (EMAPE) are used as the metrics, which are calculated as follows.

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 , \quad \text{MAE} = \frac{1}{N} \sum_{i=1}^{N} |y_i - \hat{y}_i| \tag{15}$$

$$\text{EMAPE} = \frac{1}{N} \sum_{i=1}^{N} \frac{|y_i - \hat{y}_i|}{C} \times 100 \, (\%) , \, y_i > 0.1C \tag{16}$$

where N is the number of samples, C is the solar panel capacity, y_i is the i-th ground truth value, and $\hat{y}_i$ is the i-th predicted value. As shown in the formulas, MSE penalizes larger errors more for all the samples, while MAE penalizes them equally. On the other hand, the EMAPE represents the percentage

TABLE 4. *(Continued.)* (a)-Forecasting performance comparison: MSE (Normalized scale). (b) Forecasting performance comparison: MAE (Normalized scale). (c) Forecasting performance comparison: EMAPE (%).

(c)

| Output Length (min.)	Input Length (min.)	Model										
		Comparison Group							Proposed Group			
		LSTM	Seq2seq	TF	CNN-LSTM	FF	SCINet	Dlinear	DC_LSTM_I	DC_LSTM_S	DC_TF_I	DC_TF_S
5	10	5.11	4.93	5.76	5.01	4.52	-	4.64	4.53	4.54	**4.46**	4.47
5	20	5.11	5.01	5.50	4.90	4.71	4.56	4.61	4.53	4.54	**4.46**	4.48
5	30	5.11	4.97	5.55	4.79	4.74	-	4.60	4.49	4.49	4.45	**4.43**
10	20	6.07	6.15	6.33	6.06	5.27	5.37	5.55	5.30	5.30	**5.25**	5.27
10	40	6.09	6.04	6.34	5.79	5.32	5.40	5.55	5.22	5.25	**5.20**	5.23
10	60	6.06	6.01	6.21	5.85	5.30	5.42	5.51	5.21	5.23	5.21	**5.19**
15	30	6.68	6.75	6.94	6.54	5.93	-	6.22	5.82	5.82	**5.76**	5.78
15	60	6.61	6.67	6.80	6.46	5.89	6.01	6.19	5.73	5.80	5.73	**5.72**
15	90	6.64	6.72	6.89	6.49	6.32	-	6.15	5.75	5.80	**5.72**	5.73
20	40	7.20	7.25	7.42	7.00	6.53	6.42	6.77	6.20	6.24	**6.14**	6.15
20	60	7.32	7.22	7.33	6.94	6.40	6.41	6.75	6.18	6.21	6.13	**6.12**
20	80	7.34	7.25	7.33	7.05	6.71	6.39	6.73	6.18	6.22	6.13	**6.10**
40	80	9.34	8.99	8.85	8.67	8.93	7.64	8.46	7.48	7.49	**7.37**	7.41
40	120	9.54	8.96	9.00	8.49	8.59	7.58	8.35	7.58	7.65	**7.36**	7.43
40	160	9.54	9.07	9.07	8.63	8.63	7.58	8.31	7.59	7.64	**7.34**	7.42
60	120	10.59	10.50	10.28	9.93	10.25	8.61	9.75	8.72	8.65	**8.35**	8.41
60	180	10.70	10.34	10.29	9.89	9.92	8.59	9.63	8.60	8.83	**8.30**	8.51
60	240	10.67	10.31	10.37	9.95	9.85	8.76	9.61	8.78	8.67	**8.35**	8.51
Average Loss		7.54	7.40	7.57	7.14	6.88	6.44	6.85	6.33	6.35	**6.21**	6.24
Relative Loss to the Best (%)		21.52	19.22	22.00	15.00	10.85	3.75	10.46	1.97	2.40	**Best**	0.59
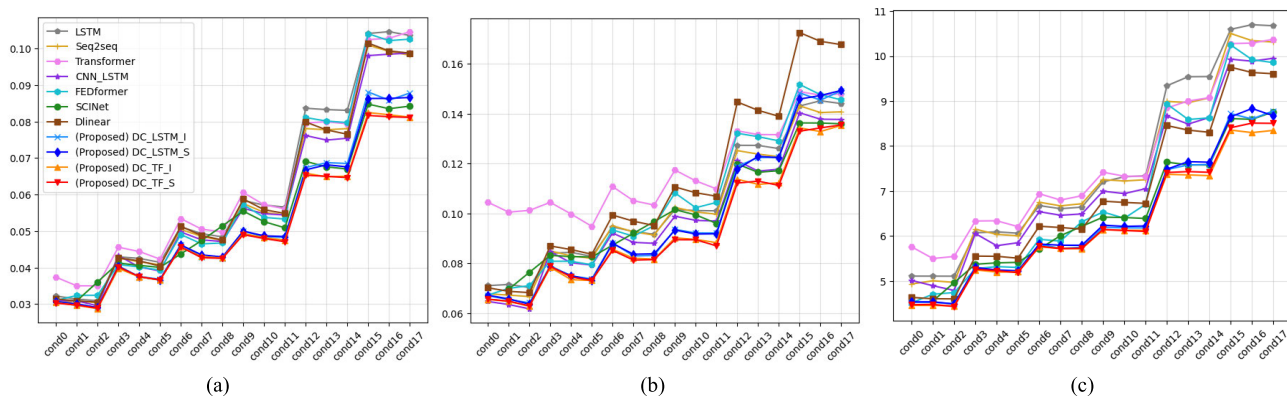
FIGURE 10. Forecasting performance comparison (Average loss): (a) MSE (Normalized scale), (b) MAE (Normalized scale), and (c) EMAPE (%).

error considering the capacity and utilization of the facility. More specifically, the EMAPE is the MAPE over the capacity calculated in the region in which a panel generates more than 10% of its capacity. In other words, the EMAPE is a valuable metric when considering the range of PV power generation that substantially impacts the power grid. Therefore, it is used as a criterion for compensation policies for renewable energy generation in South Korea. Since the solar panels at the PV site have a capacity of 3 kW, the EMAPE is calculated for the region of power generation above 0.3 kW in the experiment.

All experiments are performed on a system with an AMD EPIC 7313 CPU @ 3 GHz, 512 GB data memory, and an NVIDIA A40 48 GB GPU, and all models are implemented with the PyTorch framework. Conventional models, including the linear, LSTM, Seq2seq, attention, CNN, and

Transformer, are implemented based on the models provided by the framework. In contrast, SOTA models are implemented with source codes provided by the original authors.

B. EXPERIMENTAL RESULTS AND DISCUSSION
The detailed experimental results for each metric are shown in Table 4. For the sake of brevity, the average of the results for the three types of solar panels is presented. To observe the performance deviations according to the input and output length conditions, detailed results are provided for different historical input lengths and target output horizons within an hour. For a clear comparison, the minimum loss among the models for each length condition is bolded, and each table includes the average loss for all conditions and the loss relative to the best model. For the SCINet, the results
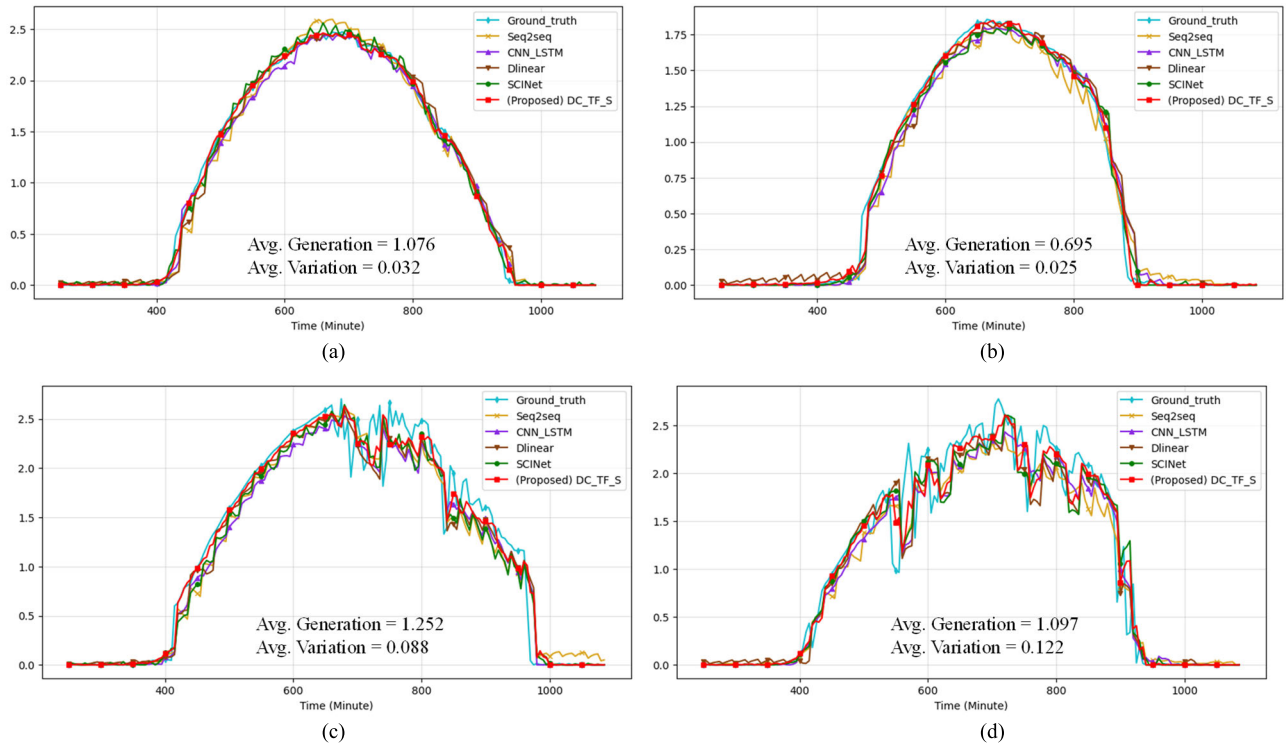
FIGURE 11. Examples of PV power generation forecasting for 1 day (kW).

for infeasible input length conditions due to its structural limitations are excluded, and its average loss is calculated by replacing the missing losses with the values from the same output length conditions.

To illustrate the performance differences more clearly, the average losses of the models are shown in Fig. 10. The order of the conditions on the horizontal axis of the graphs is the same as that in Table 4. The loss generally increases with the output length, and for the same output length, longer input sequences usually yield better results.

The experimental results demonstrate that the proposed model has the lowest loss for all metrics, i.e., the best forecasting accuracy. In particular, the proposed model configured with the Transformer achieves the best performance. As shown in Table 4 (a), the proposed models configured as DC_TF_I and DC_TF_S have the lowest MSE for all length conditions, and the performance difference becomes more pronounced as the output length increases. Based on the average loss, the SCINet and CNN_LSTM are the best performers in the comparison group, and the proposed model DC_TF_S achieves a 5.96% and 14% performance improvement over those best comparative models, respectively. The proposed models configured with the LSTM also show the second-best performance up to an output length of 40 minutes and an input length of 80 minutes, after which they perform slightly worse than the SCINet but better than the other competitors. Based on the average loss, the proposed

model DC_LSTM_S has 2.77% and 10.56% performance gains compared to the two best models in the comparison group.

Regarding the MAE, the proposed model has the lowest loss for all length conditions except when the output length is 5 minutes (next 1-point prediction). Similar to the previous evaluation with the MSE, the proposed model DC_TF_S has the highest forecasting accuracy, and the SCINet and CNN_LSTM also have the best scores among the comparison group. However, in this case, the CNN_LSTM performs slightly better than the SCINet. The LSTM-based proposed models also perform the second-best up to input and output lengths of 80 and 40 minutes, respectively, after which they fall behind the SCINet. In terms of the average MAE loss, the proposed model DC_TF_S exhibits performance gains of 5.64% and 7.13% compared to the CNN_LSTM and SCINet models, respectively, and the proposed model DC_LSTM_S also has a higher score than those best comparative models.

As indicated in Table 4 (c), the proposed model also achieves the best results based on the EMAPE metric. As a reminder, the EMAPE score is evaluated over the range of effective power generation, which makes the performance dominance of the comparison group models different from previous evaluations. In other words, the SOTA hybrid models that use the data decomposition structure rank relatively better. More specifically, the proposed model based on the Transformer performs the best under all length

conditions, and the SCINet and Dlinear models are the two top performers in the comparison group. Numerically, the proposed model DC_TF_I achieves slightly better accuracy than DC_TF_S this time, with performance gains of 3.75% and 10.46% over the SCINet and Dlinear, respectively. The proposed models with LSTM compete with the FEDformer in forecasting within 10 minutes and, similar to before, are the second-best performers until the input and output lengths are 120 and 40 minutes, respectively, after which they compete with the SCINet. To summarize, the evaluation results for all metrics demonstrate that the proposed model is the best regarding the simple average error and the error in the range considering the impact on the power grid.

In order to provide a visual representation of the performance of the models, examples of PV power generation forecasting under different weather conditions are shown in Fig. 11. The graphs show the next 20 minutes of power generation predicted from the past 40 minutes of the data at a time resolution of 5 minutes, accumulated for a day. For the sake of clarity, the results of the models that performed well in the numerical comparison are presented. In order to distinguish the weather conditions numerically, the average daily power generation and average variation of the ground truth are also attached in the figure. The average daily power generation value indicates whether the day is overall sunny or cloudy. In contrast, the average variation shows how frequently weather conditions change due to the distribution of clouds. Overcast and rainy weather conditions are excluded from the visual results because the power generation is negligible, and the performance comparison between the models is meaningless.

First, the result for the clear weather is shown in Fig. 11 (a), and the proposed model DC_TF_S best follows the ground truth. In clear weather conditions, most of the models produce patterns similar to the ground truth because of lower variability. However, the figure shows that the proposed model simulates the original most closely with the smallest overshoot and undershoot. As described in Section III, the proposed model produces the final output as a weighted sum of the outputs from the sub-models, and the dominant operation of the linear sub-model achieves this result. By similar logic, the Dlinear also accurately simulates the ground truth graph.

Then, Fig. 11 (b) shows the forecasting result in cloudy weather, where both the average daily power generation and average variation are relatively small. Even in this case, the dominance of the linear sub-model is the same as in the previous weather condition, except for the difference in scale; thus, the proposed model shows the most valid result again.

Finally, the forecasting results for the highly variable weather conditions are given in Fig. 11 (c) and (d). Because power generation frequently fluctuates, it is hard to simulate the original graph perfectly. However, the proposed model still produces outputs in the region relatively close to the ground truth. In Fig. 11 (c), the proposed model follows the original most steadily in the forenoon when the weather is relatively clear and then also forecasts the pattern in the region closer to the ground truth than the others in the afternoon, when the weather is more volatile. In other words, the proposed model can simulate PV power generation patterns that are close to the top or bottom of the original, even when the original oscillates up and down. It is also observed under the weather fluctuations throughout the day, as shown in Fig. 11 (d). To summarize, the proposed model that scores better on numerical criteria also produces more accurate results visually.

In conclusion, the experimental results demonstrate that the proposed model provides optimal ultra-short term PV power generation forecasting under different weather conditions. The results clearly show that the data decomposition structure works successfully and produces better results, particularly in the range of valid power generation. For clear weather conditions, the sub-model for the trend component dominates, and the linear sub-model in the proposed model is most effective. Meanwhile, due to frequent cloud changes, the residual component is highly irregular in 5-minute power generation patterns, making it hard to capture additional inherent periodicity (seasonality). In this regard, FEDformer, which attempts to find multiple levels of periodicity through iterative data decompositions, shows relatively poor results compared to the other SOTA models. In other words, it is preferable to determine the best processing scheme for the residual component after the initial decomposition. The experimental results prove that the Transformer and LSTM-based sub-models in the proposed model are more suitable for PV data than those of the SOTA hybrid models.

Finally, the inference time of the proposed model is measured to verify its feasibility for real-time applications. For the experiment, the sub-model for the trend is fixed with a single linear layer structure, and the processing times are measured for both cases using the Transformer and LSTM for the residual. For the longest input and output sequence length condition, i.e., forecast of future 60 minutes with past 240 minutes of data, the proposed models configured with Transformer and LSTM have average processing times of 578 us and 80 us, respectively. Even considering the specifications of the experimental system, the proposed model has reasonable computational complexity for ultra-short term forecasting at a 5-minute time resolution. Therefore, the Transformer-based proposed model can be used in most systems that require high forecasting accuracy, and the LSTM-based proposed model can be an alternative for embedded systems.

V. CONCLUSION
A novel data-driven model was proposed for intra-hour ultra-short term PV power generation forecasting. The proposed model has a hybrid structure based on the data decomposition method to achieve high prediction accuracy. The proposed model decomposes the input feature data into trend and

residual components and employs the sub-models consisting of the linear, Transformer, and LSTM to process the components.

In order to implement the proposed model, the PV site was built with three types of solar panels, and data were collected. For the experiment, environmental and electrical data were measured for 18 months, which covered different weather conditions. The proposed and comparative models were trained to predict future power generation from the historical information. Finally, their forecasting accuracies were evaluated under different input and output sequence length conditions to compare the performances of the models.

As a result, the proposed model performed best on all the evaluation criteria. The proposed model achieved 5.96%, 5.64%, and 3.75% performance improvements over the best comparison models regarding the MSE, MAE, and EMAPE, respectively. The proposed model also produced future PV power generation patterns visually closest to the ground truth under different weather conditions. In conclusion, the proposed model is the best ultra-short term forecasting model for practical PV applications and can contribute to the stable operation of power grid systems.

REFERENCES

[1] F. Wang, J. D. Harindintwali, Z. Yuan, M. Wang, F. Wang, S. Li, Z. Yin, L. Huang, Y. Fu, L. Li, and S. X. Chang, "Technologies and perspectives for achieving carbon neutrality," *Innovation*, vol. 2, pp. 1–22, Nov. 2021.

[2] R. M. Elavarasan, G. M. Shafiullah, S. Padmanaban, N. M. Kumar, A. Annam, A. M. Vetrichelvan, L. Mihet-Popa, and J. B. Holm-Nielsen, "A comprehensive review on renewable energy development, challenges, and policies of leading Indian states with an international perspective," *IEEE Access*, vol. 8, pp. 74432–74457, 2020.

[3] C. Kim, "A review of the deployment programs, impact, and barriers of renewable energy policies in Korea," *Renew. Sustain. Energy Rev.*, vol. 144, Jul. 2021, Art. no. 110870.

[4] T. Selvaraj, R. Rengaraj, G. Venkatakrishnan, S. Soundararajan, K. Natarajan, P. Balachandran, P. David, and S. Selvarajan, "Environmental fault diagnosis of solar panels using solar thermal images in multiple convolutional neural networks," *Int. Trans. Electr. Energy Syst.*, vol. 2022, pp. 1–16, Sep. 2022.

[5] J. Antonanzas, N. Osorio, R. Escobar, R. Urraca, F. J. Martinez-de-Pison, and F. Antonanzas-Torres, "Review of photovoltaic power forecasting," *Sol. Energy*, vol. 136, pp. 78–111, Oct. 2016.

[6] F. Wang, Z. Mi, S. Su, and H. Zhao, "Short-term solar irradiance forecasting model based on artificial neural network using statistical feature parameters," *Energies*, vol. 5, no. 5, pp. 1355–1370, May 2012.

[7] R. Ahmed, V. Sreeram, Y. Mishra, and M. D. Arif, "A review and evaluation of the state-of-the-art in PV solar power forecasting: Techniques and optimization," *Renew. Sustain. Energy Rev.*, vol. 124, May 2020, Art. no. 109792.

[8] Q. Shi, F. Weng, Q. Zhang, X. Wang, and F. Li, "Overvoltage mitigation through volt-VAR control of distributed PV systems," in *Proc. IEEE/PES Transmiss. Distribution Conf. Expo. (TD)*, Oct. 2020, pp. 1–5.

[9] H. Y. Yang, H. Ye, G. Wang, J. Khan, and T. Hu, "Fuzzy neural very-short-term load forecasting based on chaotic dynamics reconstruction," *Chaos, Solitons Fractals*, vol. 29, no. 2, pp. 462–469, Jul. 2006.

[10] K. B. Han, J. Jung, and B. O. Kang, "Real-time load variability control using energy storage system for demand-side management in South Korea," *Energies*, vol. 14, no. 19, p. 6292, Oct. 2021.

[11] E. Rikos, S. Tselepis, C. Hoyer-Klick, and M. Schroedter-Homscheidt, "Stability and power quality issues in microgrids under weather disturbances," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 1, no. 3, pp. 170–179, Sep. 2008.

[12] S. U. Jeon, J. Noh, S. Kang, and J.-W. Park, "Practical power management of PV/ESS integrated system," *IEEE Access*, vol. 8, pp. 189775–189785, 2020.

[13] Q. Zhang, K. Dehghanpour, and Z. Wang, "Distributed CVR in unbalanced distribution systems with PV penetration," *IEEE Trans. Smart Grid*, vol. 10, no. 5, pp. 5308–5319, Sep. 2019.

[14] G. R. Chandra Mouli, P. Bauer, T. Wijekoon, A. Panosyan, and E.-M. Bärthlein, "Design of a power-electronic-assisted OLTC for grid voltage regulation," *IEEE Trans. Power Del.*, vol. 30, no. 3, pp. 1086–1095, Jun. 2015.

[15] E. O. Hasan, A. Y. Hatata, E. A. Badran, and F. H. Yossef, "Voltage control of distribution systems using electronic OLTC," in *Proc. 20th Int. Middle East Power Syst. Conf. (MEPCON)*, Dec. 2018, pp. 845–849.

[16] Z. Zhen, S. Pang, F. Wang, K. Li, Z. Li, H. Ren, M. Shafie-Khah, and J. P. S. Catalao, "Pattern classification and PSO optimal weights based sky images cloud motion speed calculation method for solar PV power forecasting," *IEEE Trans. Ind. Appl.*, vol. 55, no. 4, pp. 3331–3342, Jul. 2019, doi: 10.1109/TIA.2019.2904927.

[17] J. Kang, J. Lee, and S. Lee, "Data-driven minute-ahead forecast of PV generation with adjacent PV sector information," *Energies*, vol. 16, no. 13, p. 4905, Jun. 2023.

[18] H. Sheng, J. Xiao, Y. Cheng, Q. Ni, and S. Wang, "Short-term solar power forecasting based on weighted Gaussian process regression," *IEEE Trans. Ind. Electron.*, vol. 65, no. 1, pp. 300–308, Jan. 2018.

[19] M. Monfared, M. Fazeli, R. Lewis, and J. Searle, "Fuzzy predictor with additive learning for very short-term PV power generation," *IEEE Access*, vol. 7, pp. 91183–91192, 2019.

[20] A. Asrari, T. X. Wu, and B. Ramos, "A hybrid algorithm for short-term solar power prediction—Sunshine state case study," *IEEE Trans. Sustain. Energy*, vol. 8, no. 2, pp. 582–591, Apr. 2017.

[21] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[22] Y. Wang, W. Liao, and Y. Chang, "Gated recurrent unit network-based short-term photovoltaic forecasting," *Energies*, vol. 11, no. 8, p. 2163, Aug. 2018.

[23] H. Zhou, Y. Zhang, L. Yang, Q. Liu, K. Yan, and Y. Du, "Short-term photovoltaic power forecasting based on long short term memory neural network and attention mechanism," *IEEE Access*, vol. 7, pp. 78063–78074, 2019.

[24] A. Vaswani, N. Shazeer, N. Parma, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. NIPS*, 2017, pp. 1–11.

[25] F. Tian, X. Fan, R. Wang, H. Qin, and Y. Fan, "A power forecasting method for ultra-short-term photovoltaic power generation using transformer model," *Math. Problems Eng.*, vol. 2022, pp. 1–15, Oct. 2022.

[26] M. Sabri and M. El Hassouni, "A novel deep learning approach for short term photovoltaic power forecasting based on GRU-CNN model," in *Proc. ES Web Conf.*, vol. 336, 2021, p. 00064.

[27] A. Agga, A. Abbou, M. Labbadi, and Y. El Houm, "Short-term self consumption PV plant power production forecasts based on hybrid CNN-LSTM, ConvLSTM models," *Renew. Energy*, vol. 177, pp. 101–112, Nov. 2021.

[28] C. Huang and M. Yang, "Memory long and short term time series network for ultra-short-term photovoltaic power forecasting," *Energy*, vol. 279, Sep. 2023, Art. no. 127961.

[29] J. Yan, L. Hu, Z. Zhen, F. Wang, G. Qiu, Y. Li, L. Yao, M. Shafie-Khah, and J. P. S. Catalao, "Frequency-domain decomposition and deep learning based solar PV power ultra-short-term forecasting model," *IEEE Trans. Ind. Appl.*, vol. 57, no. 4, pp. 3282–3295, Jul. 2021.

[30] H. H. Goh, Q. Luo, D. Zhang, H. Liu, W. Dai, C. S. Lim, T. A. Kurniawan, and K. C. Goh, "Hybrid SDS and WPT-IBBO-DNM based model for ultra-short term photovoltaic prediction," *CSEE J. Power Energy Syst.*, vol. 9, no. 1, pp. 66–76, Jan. 2023.

[31] P. Li, K. Zhou, X. Lu, and S. Yang, "A hybrid deep learning model for short-term PV power forecasting," *Appl. Energy*, vol. 259, Feb. 2020, Art. no. 114216.

[32] T. Zhou, Z. Ma, Q. Wen, X. Wang, L. Sun, and R. Jin, "FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," in *Proc. ICML*, 2022, pp. 27268–27286.

[33] M. Liu, A. Zeng, M. Chen, Z. Xu, Q. Lai, L. Ma, and Q. Xu, "SCINet: Time series modeling and forecasting with sample convolution and interaction," in *Proc. NIPS*, 2022, pp. 5816–5828.

[34] A. Zeng, M. Chen, L. Zhang, and Q. Xu, "Are transformers effective for time series forecasting?" in *Proc. AAAI*, 2023, pp. 11121–11128.

JOOSEUNG LEE received the B.S. degree in electronics engineering from Sogang University, Seoul, South Korea, in 2014, and the M.S. degree in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2016. Since 2019, he has been with the Power Grid Research Division, Korea Electrotechnology Research Institute (KERI). Prior to joining KERI, he was with Samsung Electronics, Hwaseong, South Korea, where he was involved in the research of NAND flash controllers. His current research interests include algorithms and implementations for smart grid applications and electric vehicle charging systems.

SOONWOO LEE received the Ph.D. degree in mechatronics engineering from Korea University, Seoul, South Korea, in 2018. He has been with the Korea Electrotechnology Research Institute (KERI), since 2005. He is currently a Principal Researcher with the Power ICT Center. His research interests include signal processing, digital control, and digital circuit design for power utility and smart grid applications.

JIMYUNG KANG received the B.S. and M.S. degrees in computer science from Seoul National University, Seoul, South Korea, in 2004 and 2006, respectively, and the Ph.D. degree in computer science from Sungkyunkwan University, South Korea, in 2018. Since 2006, he has been with the Power Grid Research Division, Korea Electrotechnology Research Institute (KERI). His current research interest includes machine learning in smart grids.

HUI-MYOUNG OH received the B.S. degree in electrical engineering and the M.S. and Ph.D. degrees in electrical and electronic engineering from Yonsei University, Seoul, South Korea, in 1998, 2000, and 2009, respectively. He was a Researcher with the Korea Electrotechnology Research Institute (KERI), from 2001 to 2005, and a Senior Researcher, from 2006 to 2015, where he has been a Principal Researcher, since 2016. His research interests include digital communication systems, digital twin systems, EV communication protocols, and smart grids based on renewable energy.

. . .