Unsupervised Clustering-Based Short-Term Solar Forecasting

Cong Feng, *Student Member, IEEE*, Mingjian Cui, *Senior Member, IEEE*, Bri-Mathias Hodge, *Senior Member, IEEE*, Siyuan Lu, Hendrik F. Hamann, *Member, IEEE*, and Jie Zhang, *Senior Member, IEEE*

***Abstract*—Solar forecasting accuracy is highly affected by weather conditions, therefore, weather awareness forecasting models are expected to improve the forecasting performance. However, it may not be available or reliable to classify different forecasting tasks by only using predefined meteorological weather categorization. In this paper, an unsupervised clustering-based (UC-based) solar forecasting method is developed for short-term (1-h-ahead) global horizontal irradiance (GHI) forecasting. This UC-based method consists of three parts: GHI time series unsupervised clustering, pattern recognition, and UC-based forecasting. The daily GHI time series is first clustered by an Optimized Cross-validated ClUsteRing (OCCUR) method, which determines the optimal number of clusters and best clustering results. Then, support vector machine pattern recognition is adopted to recognize the category of a certain day using the first four hours' data in the forecasting stage. GHI forecasts are generated by the most suitable models in different clusters, which are built by a two-layer machine learning based multi-model (M3) forecasting framework. The developed UC-M3 method is validated by using 1-year of data with 13 solar features from three information sources. Numerical results show that 1) UC-based models outperform non-UC (all-in-one) models with the same M3 architecture by approximately 20%; and 2) M3-based models also outperform the single-algorithm machine learning models by approximately 20%.**

***Index Terms*—Solar forecasting, unsupervised clustering, pattern recognition, machine learning, sky imaging.**

Nomenclature

A. Acronyms (Alphabetically)

AIO	All-in-one group.
ANN	Artificial neural network.
AHC, DHC	Agglomerative hierarchical clustering, divisive hierarchical clustering.
CI, CSI	Cloud index, clear sky index.
GBM	Gradient boosting machine.
GHI	Global horizontal irradiance.
ML, M3	Machine learning, machine learning based multi-model forecasting.
OCCUR	Optimal cross-validated clustering.
PR, SVM-PR	Pattern recognition, support vector machine pattern recognition.
RBR, nRBR	Red blue ratio, normalized red blue ratio.
RF	Random forest.
SAML	Single-algorithm based machine learning.
SVM, SVR	Support vector machine, support vector regression.
UC	Unsupervised clustering.

B. Variables, Indices, Parameters, Vectors, Matrices, Sets, and Functions (Alphabetically)

$\mathscr{a}, \mathscr{b}$	Indices of ML algorithm and kernel.
C	Tradeoff parameter of the SVM/SVR objective function.
CSI	Clear sky index.
$\mathscr{C}, \boldsymbol{c}, \mathscr{M}, \boldsymbol{m}$	Centroid sets, centriods, medoid sets, and medoids of clusters.
c, a, m, s	Group indices of UC, AIO, M3, and SAML groups.
d, d_f, d_t	Dimension indices.
d_a, d_b	Average distance between an object and other objects in the same cluster, average distance between an object and objects in the nearest neighboring cluster.
DNI, DHI	Direct normal irradiance, direct horizontal irradiance.
$f_{\mathscr{ab}}(\cdot), \Phi(\cdot)$	First-layer and second-layer forecasting algorithms in M3.
GHI, GHI_{clr}	Global horizontal irradiance, clear sky global horizontal irradiance.
k, k', k''	Cluster indices.
$K, K_{\max}, K_{opt}$	Total number of clusters, maximum K, optimal K.
$l, L, \mathscr{S}^{(l)}, k^{(l)}$	Hierarchical level index, total number of hierarchical levels, subset, and cluster index at hierarchical level l.

Manuscript received December 17, 2017; revised May 10, 2018 and September 23, 2018; accepted November 10, 2018. Date of publication November 15, 2018; date of current version September 18, 2019. This work was supported by the National Renewable Energy Laboratory under Subcontract No. XHQ-6-62546-01 (under the U.S. Department of Energy Prime Contract No. DE-AC36-08GO28308). Paper no. TSTE-00451-2014. *(Corresponding author: Jie Zhang.)*

C. Feng, M. Cui, and J. Zhang are with the Department of Mechanical Engineering, University of Texas at Dallas, Richardson, TX 75080 USA (e-mail: cong.feng1@utdallas.edu; mingjian.cui@utdallas.edu; jiezhang@utdallas.edu).

B.-M. Hodge is with the University of Colorado Boulder and the National Renewable Energy Laboratory, Boulder, CO 80309 USA (e-mail: BriMathias.Hodge@colorado.edu).

S. Lu and H. F. Hamann are with the IBM T. J. Watson Research Center, Yorktown Heights, NY 10598 USA (e-mail: lus@us.ibm.com; hendrikh@us.ibm.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSTE.2018.2881531

$M_{l,ij}$, $M^{i/j}$	Model with kernel l in group i and j, comparison of models in group i and j.
n, $n^{(k)}$, n_b	Total number of objects in $\mathcal{S}$ and $\mathcal{S}_k$, total number of nearest neighbours.
p, q	Indices of total cluster number, method.
r_1, r_2, r_3, $\boldsymbol{r}$	Voting indices, vector.
R, B, $nRBR$	Blue, red, and normalized red blue ratio in the RGB color system.
$\mathcal{S}$, $\mathcal{S}_k$	Universal set and clustering disjoint partitions in clustering.
T, RH, $Pres$	Temperature, relative humidity, air pressure.
$\boldsymbol{V}$, $v^{(K)}$	Vote vector, vote to total cluster number K.
WS, WD	Wind speed, wind direction.
$\boldsymbol{x}$, $\boldsymbol{x_{ij}}$, $\boldsymbol{x^{(k)}}$	Clustering objects (data vectors), jth nearest neighboring object of $\boldsymbol{x_i}$, and vectors that belong to cluster k.
$\boldsymbol{x_i^{(p)}}$, $\boldsymbol{x_i}$, $\boldsymbol{X}$	Input vectors of the pattern recognition model and forecasting models, input dataset.
$\tilde{y}$, $\hat{y}$, $y_i^{(p)}$	Values of the first-layer forecasts, second-layer final forecasts in M3, output of the pattern recognition model.
$\tilde{\boldsymbol{Y}}$, $\hat{\boldsymbol{Y}}$	Vectors of the first-layer forecasts, second-layer final forecasts.
α, ψ	Weighted vector, bias constant.
β	Connectedness measurement.
$w_{i,k}$	Membership of $\boldsymbol{x_i}$ in cluster k.
$\|\cdot\|_2$	Euclidean norm.
$\kappa(\cdot)$, ϱ	Kernel function, kernel parameter of the SVM-PR model.
ξ, ξ^*	Upper and lower bounds of the deviations around SVM objective function.
μ, σ, H	Mean, standard deviation, Rényi entropy of nRBR.
$\boldsymbol{\mu}$, $\boldsymbol{\sigma}$, $\boldsymbol{H}$	Set of mean, set of standard deviation, set of Rényi entropy of nRBR.

C. Evaluation Metrics (Alphabetically)

$Conn$	Connectivity index.
$Silh$	Silhouette width.
$Dunn$	Dunn's index.
S_{tv}, P_{cs}, A_{cc}	Pattern recognition sensitivity, precision, overall accuracy.
$nMAE$	Normalized mean absolute error.
$nRMSE$	Normalized root mean square error.
$ImpA$	Improvement of $nMAE$.
$ImpR$	improvement of $nRMSE$.

I. Introduction

SOLAR power is a potential alternative to fossil fuel-generated power due to its sustainability. The global installed photovoltaic (PV) capacity is expected to reach 4,600 GW by 2050, providing approximately 16% electricity worldwide [1]. The U.S. has installed 47 GW of PV by 2017, with California having the highest solar penetration [2]. However, the variability and uncertainty in PV power pose a number of challenges to power system operations. Accurate solar forecasting (including solar power and solar irradiance forecasting) could assist power system operators better manage the uncertainties and reduce risks, especially under high solar penetration scenarios.

A collection of statistical and machine learning (ML) methods have been proposed in the literature for short-term solar forecasting. For example, Shakya *et al.* [3] developed a 1-day-ahead (1DA) solar irradiance forecasting model based on Markov swiching method, which generated solar forecasting for remote areas. Zhang *et al.* [4] compared radial basis function neural networks, least square support vector machine (SVM), k-nearest neighbor (kNN), weighted kNNs (WkNNs), and found that kNN and WkNNs yielded the most competitive forecasting results. A comprehensive review of these methods can be found in latest review papers [5]–[7]. Even though the learning ability of ML models has been enhanced notably, it is still challenging to capture the complex input-output relationship with single-algorithm based ML (SAML) methods, especially under different conditions [8]. For example, none of SAML models outperformed others under all weather conditions in [9]. On the other hand, the forecasting performance of ML models is critically influenced by the inputs. Some advanced techniques have been explored recently to enhance ML forecasting by providing informative input features, such as total sky images [10], [11], satellite images [12], ground-based sensor measurements [13], and numerical weather predictions [14]. Among these information sources, features such as historical forecasting variable, cloud index (CI), red blue ratio (RBR) features of sky images and ground-based weather measurements are among the most informative inputs to the ML models for short-term solar forecasting.

Solar features are highly influenced by weather conditions. Therefore, it is challenging to get accurate forecasts under different weather conditions from a single model. In order to divide time series forecasting into different conditions where disparate models can be applied, two processes are required: *clustering* and *classification*. *Clustering* is an unsupervised process to distinguish and label the type of each time period in the training data. *Classification* is to identify the category of a time period in the forecasting stage in a supervised manner. Several clustering methods have been reported in the literature to divide a forecasting task into subtasks. For example, a combination of self orgnizing map and learning vector quantization was used in [15] to distinguish three predefined weather types. K-means clustering was applied in [16] to cluster solar irradiance patterns. A pattern discovery method was adopted in [17] to classify different PV system classes. A more comprehensive review of clustering and classification methods in the renewable energy domain can be found in [18]. Nevertheless, several drawbacks exist in these methods: (i) most of the existing work uses pre-defined or meteorologically defined criteria (such as weather condition) in the clustering process, which may not be available or reliable for forecasting methods; (ii) the number of clusters is not optimized for clustering; (iii) adopted clustering methods are not always reliable for data with varying characteristics.

Pattern recognition (PR) is a kind of classification techniques that identify labels of objects. In solar forecasting, PR has been adopted to recognize to which cluster a forecasting object belongs, therefore a suitable model can be selected to perform the forecasting. For example, the forecasting error at the current time was used to identify the current pattern and select the corresponding model in the next forecasting step in [19]. The temperature difference between the forecast day and the current day was employed to identify the weather type in [20]. An SVM model was used to determine weather types using six extracted solar features in [21]. However, existing methods have several nonnegligible limitations: (i) PR is mainly used in 1 day-ahead (DA) or longer time horizon forecasting, which takes advantage of longer input vectors and therefore is theoretically easier than that for shorter-term time horizons; (ii) some models require indirect variables, such as temperature and clear sky index (CSI), to determine the weather pattern; (iii) more advanced algorithms are required to improve the PR accuracy.

To address the aforementioned limitations, in this paper we seek to improve solar forecasting by enhancing solar data clustering, PR, and forecasting learning abilities simultaneously. In what follows, an advanced unsupervised clustering (UC) method is developed, which only utilizes GHI time series without other indirect variable information. Then, PR identifies the cluster to which a forecasting day belongs with first few hours' data. Lastly, a two-layer Machine Learning based Multi-Model (M3) forecasting framework [22] is developed to reinforce learning abilities of the ML models. The main innovations and contributions of this paper include:

i) Developing a novel Optimized Cross-validated ClUsteRing (OCCUR) method to optimize both the number of clusters and the clustering performance;
ii) Adopting an advanced SVM PR (SVM-PR) method to identify categories of forecasting days with a small number of inputs;
iii) Leveraging the powerful learning ability and robustness of a two-layer M3 model in the forecasting stage;
iv) Validating the superiority of the developed UC-M3 based method by exploring the effectiveness of both UC and M3 methods under different conditions that consider both calendar and clustering effects.

The remainder of the paper is organized as follows. The OCCUR method is developed in Section II. Section III describes the SVM-PR, M3, and the overall unsupervised clustering based (UC-based) solar forecasting method (denoted as UC-M3). Numerical simulations are carried out in Section IV to validate the developed UC-M3 method. Section V summarizes the conclusions and discusses the future work.

II. Optimized Cross-validated ClUsteRing (OCCUR)

Clustering unlabelled daily GHI time series is an unsupervised learning problem, wherein the inherent structure needs to be deduced. Unsupervised learning is far more challenging than supervised learning due to the lack of data foreknowledge. The number of clusters also varies with different UC algorithms and evaluation metrics. In this paper, an Optimized Cross-validated ClUsteRing (OCCUR) method is developed to optimize and cross-validate the number of clusters using UC algorithms. The OCCUR method adopts multiple UC algorithms to perform clustering independently. The clustering results are cross-validated using several internal validity indices.

A. Unsupervised Clustering (UC) Algorithms

Four UC algorithms are used in the OCCUR method, which are: K-means, K-medoids, Agglomerative hierarchical clustering (AHC), and divisive hierarchical clustering (DHC). K-means is a widely used UC algorithm. Given a dataset $\mathscr{S} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ of n d-dimensional vectors, K-means is a partitional clustering method to construct K disjoint subsets $\mathscr{S} = \{\mathscr{S}_1, \ldots, \mathscr{S}_K\}$, such that $\mathscr{S}_k \neq \varnothing$ ($k = 1, 2, \ldots, K$), $\mathscr{S}_k \bigcap \mathscr{S}_{k'} = \varnothing$ ($k, k' = 1, 2, \ldots, K$ and $k \neq k'$), and $\bigcup_{k=1}^{K} \mathscr{S}_k = \mathscr{S}$ [23]. The main idea of the algorithm is to determine the centroids, $\mathscr{C} = \{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_K\}$, and disjoint subsets $\mathscr{S}$ as follows:

$$w_{i,k} = \begin{cases} 1, & \boldsymbol{x}_i \in \mathscr{S}_k \\ 0, & \boldsymbol{x}_i \notin \mathscr{S}_k \end{cases} \tag{1}$$

$$\boldsymbol{c}_k = \frac{\sum_{i=1}^{n} w_{i,k} \boldsymbol{x}_i}{\sum_{i=1}^{n} w_{i,k}} \tag{2}$$

$$\mathscr{S}_k = \{\boldsymbol{x}^{(k)}\} = \left\{\boldsymbol{x}_1^{(k)}, \ldots, \boldsymbol{x}_{n^{(k)}}^{(k)}\right\} \tag{3}$$

where $w_{i,k}$ is the data vector membership of $\boldsymbol{x}_i$ in cluster k. For example, $w_{1,1} = 1$ means $\boldsymbol{x}_1$ belongs to $\mathscr{S}_1$ and $w_{1,1} = 0$ means $\boldsymbol{x}_1$ does not belong to $\mathscr{S}_1$. $\boldsymbol{x}^{(k)}$ is the data vector categorized into cluster k. The K-means algorithm repeats iterative refinement steps by updating the centroids and subsets based on Eqs. (2) and (3), until reaching the optima given by [23], [24]:

$$\underset{\mathscr{S}}{\operatorname{argmin}} \sum_{k=1}^{K} \sum_{i=1}^{n} w_{i,k} \left\| (\boldsymbol{x}_i - \boldsymbol{c}_k) \right\|_2 \tag{4}$$

where $\|\cdot\|_2$ is the Euclidean norm, which is used to calculate the distance. Note that the distance can be modified to meet the requirements of other research objectives (e.g., correlation-based distance can be used in temporal-spatial clustering).

Another partitional UC algorithm adopted is K-medoids. Instead of clustering based on the centroids, K-medoids seeks the medoids of clusters. A medoid is the most centrally located object (data vector in the regression case) within a cluster, which makes K-medoids more robust than the K-means in some cases. Medoids, $\mathscr{M} = \{\boldsymbol{m}_1, \ldots, \boldsymbol{m}_K\}$, are determined by minimizing the summed distance of a data vector to other vectors within the same cluster [25]:

$$\boldsymbol{m}_k = \underset{\boldsymbol{x} \in \mathscr{S}_k}{\operatorname{argmin}} \sum_{i=1}^{n} \sum_{j=1}^{n} w_{i,k} w_{j,k} \left\| (\boldsymbol{x}_i - \boldsymbol{x}_j) \right\|_2 \tag{5}$$

where $\boldsymbol{m}_k$ is the medoid of the cluster k. The objective function of the K-medoids method is modified as:

$$\underset{\mathscr{S}}{\operatorname{argmin}} \sum_{k=1}^{K} \sum_{i=1}^{n} w_{i,k} \left\| (\boldsymbol{x}_i - \boldsymbol{m}_k) \right\|_2 \tag{6}$$

where $\mathscr{S}$ and $\mathscr{M}$ are updated in each iteration until the convergence condition is satisfied.

Agglomerative hierarchical clustering (AHC) is a bottom-up unsupervised hierarchical clustering method. Compared with partitional methods, a pre-defined cluster number K is not required in AHC [26]. AHC constructs the hierarchy by merging the most similar pairs of lower-level nodes from the bottom to the top. In this paper, the distance of two clusters is calculated by the average linkage method, which is defined as the averaged pairwise distance between data vectors from the two clusters [27]:

$$\mathscr{S}^{(l)} \rightarrow \mathscr{S}^{(l+1)}: \\ \operatorname{argmin}_{\mathscr{S}^{(l+1)}} \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} w_{i,k^{(l)}} w_{j,k'^{(l)}} \left\|(\boldsymbol{x}_i - \boldsymbol{x}_j)\right\|_2}{\sum_{i=1}^{n}\sum_{j=1}^{n} w_{i,k^{(l)}} w_{j,k'^{(l)}}}, \ k \neq k' \tag{7}$$

where $\mathscr{S}^{(l)} = \{\mathscr{S}_1^{(l)}, \ldots, \mathscr{S}_{K^{(l)}}^{(l)}\}$ is the clustering set at hierarchical level l ($1 \leq l \leq L-1$, where 1 is the bottom level and L is the top level). $w_{i,k^{(l)}}$ is the membership of $\boldsymbol{x}_i$ in cluster $\mathscr{S}_{k^{(l)}}^{(l)}$. k and k' ensure that the average linkage method is applied to two different clusters at a certain hierarchical level. The clustering result is obtained by cutting the hierarchical dendrogram at a certain height.

Another hierarchical clustering method is divisive hierarchical clustering (DHC) [28], which constructs the hierarchy in a top-down manner. DHC splits a cluster into two subclusters until only singletons are left. In the splitting process, the bipartitions are determined by maximizing the between-subcluster dissimilarity:

$$\mathscr{S}^{(l+1)} \rightarrow \mathscr{S}^{(l)}: \\ \operatorname{argmax}_{\mathscr{S}^{(l)}} \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} w_{i,k^{(l)}} w_{j,k'^{(l)}} \left\|(\boldsymbol{x}_i - \boldsymbol{x}_j)\right\|_2}{\sum_{i=1}^{n}\sum_{j=1}^{n} w_{i,k^{(l)}} w_{j,k'^{(l)}}}, \ k \neq k' \tag{8}$$

where parameters have the same meaning as Eq. (7). The same strategy in AHC is used to obtain the clustering results. The complete enumeration splitting is adopted for the optimum in this paper, which can be found in [29].

B. Clustering Assessment Metric

Evaluating the clustering correctness is challenging due to the absence of data labels. Satisfactory clustering is expected to have desirable *connectedness* among clustering objects, *cohesion* (also known as compactness or homogeneity) within every cluster, and *separation* between clusters. To assess the clustering performance of the aforementioned UC methods, three internal validity indices are adopted to quantify the clustering performance from different perspectives [30]–[33].

Connectivity, $Conn$, measures the connectedness between an object and its nearest neighbors, which is expressed as:

$$\beta_{\boldsymbol{x}_i,\boldsymbol{x}_{ij}} = \begin{cases} \frac{1}{j}, & \boldsymbol{x}_i, \boldsymbol{x}_{ij} \in \mathscr{S}_k \\ 0, & \boldsymbol{x}_i \in \mathscr{S}_k, \boldsymbol{x}_{ij} \notin \mathscr{S}_k \end{cases} \tag{9}$$

$$\text{Conn} = \sum_{i=1}^{n}\sum_{j=1}^{n_b} \alpha_{\boldsymbol{x}_i,\boldsymbol{x}_{ij}} \tag{10}$$

where $\boldsymbol{x}_{ij}$ is the jth nearest neighbour of $\boldsymbol{x}_i$. $\beta_{\boldsymbol{x}_i,\boldsymbol{x}_{ij}}$ is the connectedness measurement between $\boldsymbol{x}_{ij}$ and $\boldsymbol{x}_i$. n_b is the size of the nearest neighboring objects. $k = 1, \ldots, K$ is a subset index. A smaller $Conn$ value indicates better clustering performance ($Conn \in (0, +\infty)$).

Silhouette width, $Silh$, quantifies both clustering cohesion and separation. $Silh$ is the average of the Silhouette coefficients of all objects. Silhouette coefficients are calculated based on the distance between a clustering object and other objects within the same cluster, and the distance between the same object and objects in the nearest neighboring cluster. It is expressed as:

$$d_a(i) = \frac{\sum_{j=1}^{n} w_{i,k} w_{j,k} \left\|(\boldsymbol{x}_i - \boldsymbol{x}_j)\right\|_2}{\sum_{j=1}^{n} w_{j,k}} \tag{11}$$

$$d_b(i) = \frac{\sum_{j=1}^{n} w_{i,k} w_{j,k'} \left\|(\boldsymbol{x}_i - \boldsymbol{x}_j)\right\|_2}{\sum_{j=1}^{n} w_{j,k'}} \tag{12}$$

$$Silh = \frac{1}{n}\sum_{i=1}^{n} \frac{d_b(i) - d_a(i)}{\max(d_a(i), d_b(i))} \tag{13}$$

where $Silh \in [-1, +1]$. $Silh = +1$ indicates desired clustering, vice versa.

The Dunn's index, $Dunn$, is also able to measure both the cohesion and separation of a clustering result, by a ratio between the minimal inter-cluster distance to the maximal intra-cluster distance.

$$Dunn = \frac{\min\left\{\sum_{i=1}^{n}\sum_{j=1}^{n} w_{i,k} w_{j,k'} \left\|(\boldsymbol{x}_i - \boldsymbol{x}_j)\right\|_2\right\}}{\max\left\{\sum_{i=1}^{n}\sum_{j=1}^{n} w_{i,k''} w_{j,k''} \left\|(\boldsymbol{x}_i - \boldsymbol{x}_j)\right\|_2\right\}} \tag{14}$$

where k, k', and k'' ensure the independency of the clusters. $Dunn \in [0, +\infty)$, and a larger $Dunn$ value indicates better clustering performance.

C. Cross-Validation Process

The developed Optimized Cross-validated ClUsteRing (OCCUR) method optimizes the number of clusters, which is expected to be more accurate and reliable than that determined by a single UC method. The optimal cluster number is determined by cross-validating the clustering performance of several UC methods from various perspectives. OCCUR is expected to avoid drawbacks of a single clustering method and find the optimum. The pseudocode of the OCCUR method is illustrated in Algorithm 1. The aforementioned four UC methods are adopted to cluster the time series data into a predefined cluster number K ($K = 2, \ldots, K_{\max}$), which is evaluated by the three internal metrics mentioned above. The clustering results with smaller

Algorithm 1: Optimized Cross-validated ClUsteRing (OCCUR) method.

Initialize voting score vector $\boldsymbol{V} = \{v^{(2)}, ..., v^{(K_{max})}\}$
for $p \leftarrow 1$ **to** $(K_{max} - 1)$ **do**
  for $q \leftarrow 1$ **to** 4 **do**
    Cluster using the qth method from Eqs. 1 - 8, with $(p+1)$ clusters: $\rightarrow \mathcal{S}_q^p$
    Assess clustering performance based on $\mathcal{S}_\alpha^q$ by Eqs. 9 - 14: $\rightarrow Conn_{pq}$, $Silh_{pq}$, $Dunn_{pq}$
  end
end
for $q \leftarrow 1$ **to** 4 **do**
  Construct evaluation vectors: $\boldsymbol{Conn_q} = \{Conn_{pq}\}$, $\boldsymbol{Silh_q} = \{Silh_{pq}\}$, $\boldsymbol{Dunn_q} = \{Dunn_{pq}\}$.
  Initialize dynamic evaluation vectors: $\boldsymbol{Conn'_q} = \boldsymbol{Conn_q}$, $\boldsymbol{Silh'_q} = \boldsymbol{Silh_q}$, $\boldsymbol{Dunn'_q} = \boldsymbol{Dunn_q}$
  for $v \leftarrow 1$ **to** $(K_{max} - 1)$ **do**
    Obtain the voting index by sorting evaluation vectors: $r_1 = \underset{p}{\operatorname{argmin}}\, \boldsymbol{Conn'_q}$, $r_2 = \underset{p}{\operatorname{argmax}}\, \boldsymbol{Silh'_q}$, $r_3 = \underset{p}{\operatorname{argmax}}\, \boldsymbol{Dunn'_q}$
    Vote the cluster number based on three evaluation metrics: $v^{(r+1)} = K_{max} - v + v^{(r+1)}$, $\boldsymbol{r} = \{r_1, r_2, r_3\}$
    Update the dynamic evaluation metrics by eliminating $Conn_{r_1 q}$, $Silh_{r_2 q}$, $Dunn_{r_3 q}$
  end
end
Obtain the optimal cluster number: $K_{opt} = \underset{K}{\operatorname{argmax}}\, \boldsymbol{V}$

$Conn$ and larger $Silh$/$Dunn$ values will receive more votes. The optimal cluster number $K_{\max}$ is the K with the most votes. The result from the best model is selected as the final clustering result and used in the following PR and forecasting stages.

III. Pattern Recognition and Clustering-Based Forecasting

A. SVM Pattern Recognition

After determining the clusters by the OCCUR method, it is also challenging to identify which cluster the forecasting day belongs to. As shown in Fig. 1 by using the direct classification method, the more hours of data used to determine the cluster label of a forecasting day, the more accurate the classification is. The classification accuracy also highly affects the forecasting performance. In this case, too many hours (of a day) of data are needed to achieve a satisfying classification and forecasting accuracy (i.e., 11 hours' of data is required to achieve a 80% accuracy), which is impractical in short-term forecasting. Thus, an advanced classification method, SVM PR (SVM-PR), is applied to identify the data category of a day by only using the first few hours' data (i.e., 4 hours data in this paper).

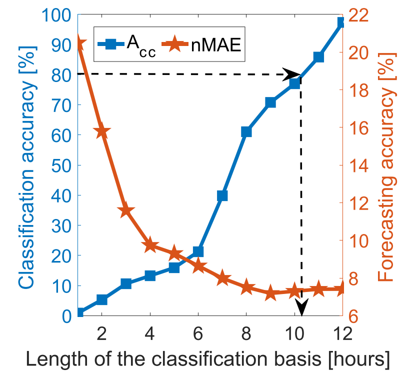


Fig. 1. Classification and forecasting accuracy by using a direct classification method. The direct classification method and the support vector regression (SVR) method are adopted in the classification and forecasting, respectively [11]. The overall accuracy ($A_{\rm cc}$) and the normalized mean absolute error ($nMAE$) are evaluation metrics to measure the classification and forecasting accuracy, respectively. A larger $A_{\rm cc}$ and a smaller $nMAE$ indicate a better classification and more accurate forecasting, respectively.

SVM-PR is a classification-based method, which is trained with labeled data and identifies labels of the forecasting data. To model an SVM classifier, the outputs are assumed to take a form of [21]:

$$y_i^{(p)} = \alpha_i^T \cdot \kappa(\boldsymbol{x}_i^{(p)}, \boldsymbol{x}_i'^{(p)}) + \psi \tag{15}$$

where $y_i^{(p)}$ and $\boldsymbol{x}_i^{(p)}$ are the output (data cluster label) and $d_x^{(p)}$-dimensional ($d_x^{(p)} = d_f^{(p)} \times d_t^{(p)}$, where $d_f^{(p)}$ is the number of features in the PR, and $d_t^{(p)}$ is the number of hours chosen as classification basis) input vector of the SVM-PR model. α_i is a $d_l^{(p)}$-dimensional weighted vector. ψ is a bias constant. $\kappa(\cdot)$ is a kernel function that maps the $d_x^{(p)}$-dimensional input vector into a $d_l^{(p)}$ feature space. A radial basis function (RBF) is selected as the kernel function, expressed as:

$$\kappa(\boldsymbol{x}, \boldsymbol{x}') = e^{-\frac{\|\boldsymbol{x}-\boldsymbol{x}'\|}{2\varrho^2}} \tag{16}$$

where ϱ is a kernel parameter. The objective function of SVM-PR is formulated as:

$$\min \quad \frac{1}{2}\|\alpha\|^2 + C\left(\sum_{i=1}^{t}(\xi_i + \xi_i^*)\right) \tag{17}$$

subject to:

$$\langle \alpha, \boldsymbol{x}_i \rangle + \psi - y_i \leq \epsilon + \xi_i^*, \ \forall i \tag{18a}$$

$$y_i - \langle \alpha, \boldsymbol{x}_i \rangle - \psi \leq \epsilon + \xi_i, \ \forall i \tag{18b}$$

$$\xi_i, \ \xi_i^* \geq 0 \tag{18c}$$

where ξ and ξ^* are the upper and lower ϵ bounds of deviations around the objective function, respectively. C is a tradeoff parameter. Once the classfier model is trained, the data cluster label can be recognized by an inputs vector $\boldsymbol{x}$ with the same features.

B. Machine Learning Based Multi-Model (M3) Forecasting

M3 is a two-layer ML based method for short-term forecasting, as shown in the brown box of Fig. 2. Multiple ML

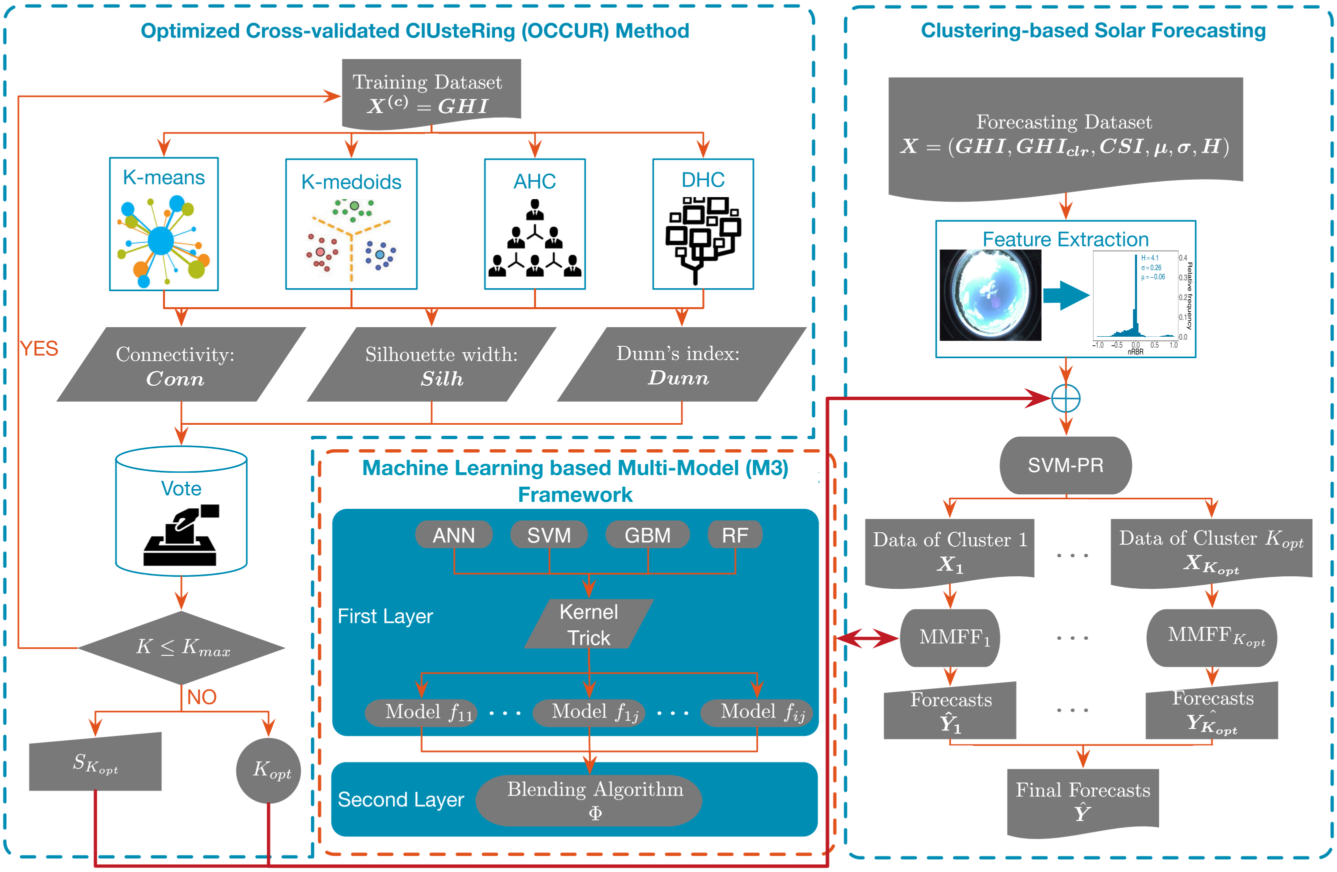


Fig. 2. Overall framework of the unsupervised clustering-based short-term solar forecasting method.

algorithms with several kernels generate forecasts, $\tilde{Y}$, independently in the first layer. Then the forecasts are blended by a ML algorithm in the second layer, which gives the final forecasts, $\hat{Y}$. ML algorithms used in M3 include artificial neural networks (ANN), SVR, gradient boosting machines (GBM), and random forests (RF). M3 has been shown to perform better than SAML methods in wind [34], [35], solar, and load forecasting [36]. M3 can be expressed as:

$$\tilde{y}_{i,ab} = f_{ab}(\boldsymbol{x}_i) \tag{19}$$

$$\hat{y}_i = \Phi(\tilde{\boldsymbol{y}}_i) \tag{20}$$

where i is the time index, $f_{ab}(\cdot)$ is the model in the first-layer using ath ML algorithm with kernel b, $\tilde{y}_{ab}$ is the forecast provided by model f_{ab}, $\boldsymbol{x}_i \in \boldsymbol{X}$ is the input vector to the first-layer models, $\tilde{\boldsymbol{y}} = \{\tilde{y}_{ab}\}$ is the combination of the first-layer forecasts, $\hat{y}_i$ is the final forecast at time i, and $\Phi(\cdot)$ is the blending algorithm in the second layer. Note that several blending algorithms can be applied in the second layer, and the best performing M3 model (with the most accurate blending algorithm) in each cluster is selected to construct the final forecasting framework (denoted as C_{opt}). This training process is evaluated through a 10-fold cross-validation. More details of M3 can be found in [22].

C. Clustering-Based Solar Forecasting

The UC-M3 solar forecasting framework integrates OCCUR clustering, SVM-PR, and M3, as shown in Fig. 2. The optimal cluster number K_{opt} and the best clustering result $S_{K_{opt}}$ are first determined by OCCUR using the training dataset (only use everyday's GHI). Then, SVR-PR is modeled by labeled $S_{K_{opt}}$, which is adopted to recognize the category of a certain day using the first 4 hours' data (from 7 am to 10 am, including all the solar features) in the forecasting dataset. M3 is used as the forecasting engine, which is built for each cluster separately. Since most GHIs before 7 am are close to zero, which do not provide enough information to build an efficient learning model, GHIs at 7 am are forecasted by a 1-day-ahead (1DA) persistence of cloudiness model. This 1DA persistence of cloudiness model assumes a constant clear sky index (CSI) within 24 hours, which is expressed by:

$$GHI_p(t+\Delta t) = \frac{GHI(t)}{GHI_{clr}(t)} \times GHI_{clr}(t+\Delta t) \tag{21}$$

where $GHI_p(t+\Delta t)$ means the GHI persistent prediction at time $t+\Delta t$; GHI and GHI_{clr} are GHI measurements and clear-sky GHI values, respectively; Δt is the forecasting time horizon of the persistence of cloudiness method, which is 24 in this model. The GHIs at 8 am, 9 am, and 10 am are forecasted by 3 hourly-similarity based M3 models. For example, the $\mathrm{M3}_{8\,\mathrm{am}}$ model is trained by $\{\mathbf{GHI}_{\mathbf{7\,am}} | \mathbf{GHI}_{\mathbf{8\,am}}\}$. More details about hourly-similarity based solar forecasting can be found in [10] and [11]. Note that several blending algorithms can be applied in the second layer, and the best M3 model, $\mathrm{M3}_{k_a}$, with a certain blending algorithm in each hour/cluster is selected to construct the final forecasting framework. This training process is evaluated through a 10-fold cross-validation.

IV. Case Study

A. Data Summary and Feature Extraction

To obtain well-performing data-driven models, suitable features need to be extracted from different information sources and fed into the models. The features selected in this paper are from three information sources: (i) GHI features: historical GHI (GHI), clear sky GHI (GHI_{clr}), and clear sky index (CSI); (ii) sky imaging features: mean (μ), standard deviation (σ), and Rényi entropy (H) of the normalized sky image pixel RBR ($nRBR$) values; and (iii) other meteorological measurements: direct normal irradiance (DNI), direct horizontal irradiance (DHI), temperature (T), relative humidity (RH), pressure (Pres), wind speed (WS), and wind direction (WD).

GHI_{clr} is the GHI value under cloudless conditions, which is generated by a clear-sky model. In this paper, the Ineichen and Perez model [37] is selected as the clear-sky model. CSI is the ratio of GHI and GHI_{clr}. The final three features are extracted by sky image processing, and the $nRBR$ of a pixel is calculated by:

$$nRBR_i = \frac{R_i - B_i}{R_i + B_i} \tag{22}$$

where R_i and B_i represent the red and blue values of the ith sky image pixel in the RGB color system, respectively. The number of pixels in each image is 1,392 × 1,040. $nRBR$ is the basis to calculate the three sky imaging features μ, σ, and H. H is the Rényi entropy, defined as:

$$H = \frac{1}{1-\gamma} \log \left[\sum_{i=1}^{n} (p_i^\gamma) \right] \tag{23}$$

where $\gamma = 2$ is the order of Rényi entropy. p_i^γ is the frequency for the ith bin (out of 150 evenly spaced bins). These 13 features (i.e., $GHI, GHI_{clr}, CSI, \mu, \sigma, H, DNI, DHI, T, RH, Pres$, WS, and WD) compose the feature space to serve as the inputs to the PR model.

An 1-year hourly GHI and sky imaging dataset released by the National Renewable Energy Laboratory (NREL) is adopted in the case study, which was collected at a location in Colorado (latitude = 39.74° North, longitude = 105.18° West, elevation = 1,828.8 m). Since solar feature time series has strong seasonal patterns (the strength of seasonality [38] of GHI is 0.84 out of 1), the training data is randomly selected from each month, and the remaining data is used for testing. The ratio of training days to testing days is 3:1. We assume that by randomly partitioning days into training or testing datasets, the model generality can be better assessed. This data partitioning strategy has been widely used in power system time series forecasting, such as Global Energy Forecasting Competition (GEFCom) 2012 [39] and GEFCom 2014 [40]. The GHIs at early morning (before 7 am) and late night (after 7 pm) are not included in this paper, since most GHI values during this period are zero.

TABLE I
Computational Time (min)

Process	Training Time	PR / Forecasting Time
OCCUR	3.08×10^{-4}	
SVR-PR	2.12	2.12×10^{-4}
UC-M3 forecasting	8.63	6.81×10^{-2}
AIO-M3 forecasting	5.78	4.00×10^{-2}
UC-SAML forecasting	3.66	5.70×10^{-2}
AIO-SAML forecasting	2.87	2.28×10^{-2}

To validate the developed UC-M3 solar forecasting method, the effectiveness of both UC-based forecasting and M3-based forecasting are evaluated by comparing two sets of counterparts, which are M3 models vs. SAML models and UC-based models vs. all-in-one (AIO) models. Therefore, there are four groups of models built in this research, which are listed as:

- *Group 1 (the developed models):* UC and M3 (UC-M3) based solar forecasting, which clusters forecasting tasks by OCCUR and adopts M3 as the forecasting engine.
- *Group 2:* UC and SAML (UC-SAML) based solar forecasting, which clusters forecasting tasks by OCCUR and adopts SAML models as forecasting engines.
- *Group 3:* AIO and M3 (AIO-M3) based solar forecasting, which does not cluster forecasting tasks and adopts M3 as the forecasting engine.
- *Group 4:* AIO and SAML (AIO-SAML) based solar forecasting, which does not cluster forecasting tasks and adopts SAML models as forecasting engines.

In each of the above four groups, several ML algorithms with multiple kernels are adopted to test the generality of the developed UC-based forecasting method. Details of these algorithms can be found in [10]. The experiment is carried out on a laptop with an Intel Core i7 2.6 GHz processor and a 16.0 GB RAM, and the computational time is summarized in Table I. The time of forecasting model training varies significantly. UC-based models and M3-based models need more time for training than AIO models and SAML models. This is because the UC method has more forecasting models and M3 has two layers. Specifically, the developed UC-M3 method requires 8.63 mins for training based on 0.75 year of data and needs 2.12×10^{-4} mins to generate 0.25 year of forecasts. The computational time of the developed method is desirable for 1HA forecasting.

B. OCCUR Clustering Results

The OCCUR method is first carried out to determine the optimal number of clusters. Fig. 3 shows the clustering performance with $K_{\max} = 14$. Generally, the connectedness, cohesion, and separation of clustering deteriorate with the increasing number of clusters. When the number of clusters is small ($K \leq 4$), different UC methods illustrate almost equivalent clustering power. With the increasing number of clusters, the clustering goodnesses of different methods become distinctive. Fig. 3 also shows contrary results when evaluated by different internal metrics. For example, the cohesion and separation of the clustering are satisfying but the connectedness is undesirable when $K = 2$, compared to $K = 3$ and 4. The number of clusters $K = 3$ is

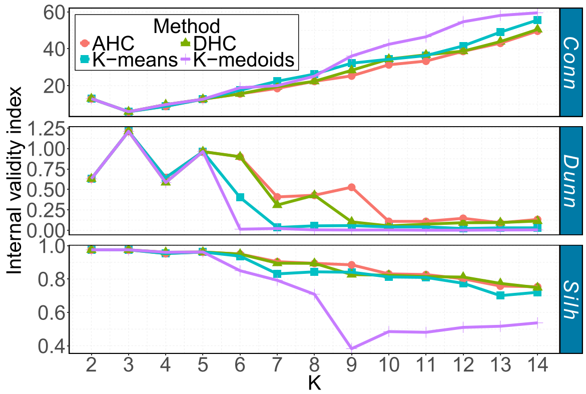
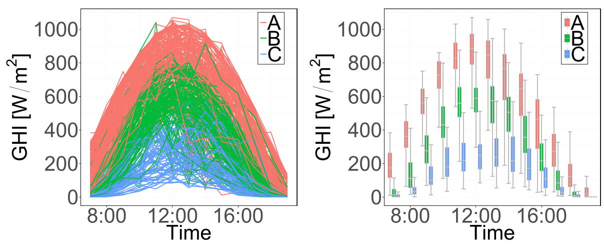


Fig. 3. OCCUR clustering results. $K_{\max} = 14$, $K_{opt} = 3$.



(a) Clustered GHI time series (b) Clustered GHI statistics

Fig. 4. OCCUR clustering results.

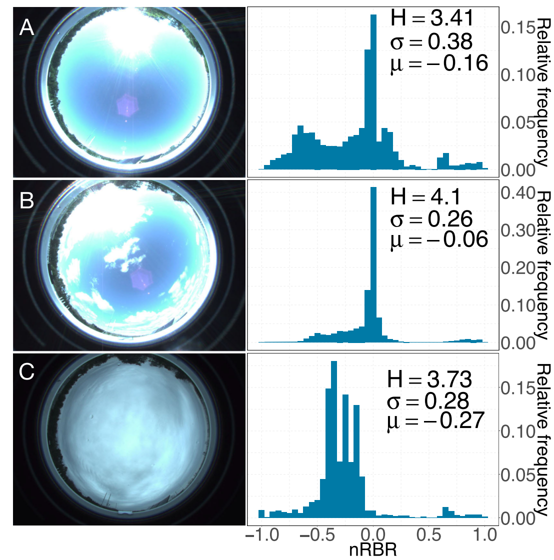


Fig. 5. Sky images and corresponding $nRBR$ distributions of 3 clusters.

more suitable than $K = 2$ and 4 based on $Dunn$; but metrics of $Conn$ and $Silh$ show a different trend. Overall, the optimal number of clusters is $K_{opt} = 3$, which is determined by the OCCUR voting process in Algorithm 1. The best UC method with $K = 3$ is AHC ($Conn = 5.86$, $Dunn = 1.21$, $Silh = 0.98$), which is adopted to cluster the training data. The clustered daily GHI time series and corresponding statistics are illustrated in Fig. 4. The clustering is evidently layered, which indicates successful clustering.

C. Pattern Recognition Results

At the forecasting stage, the category of a certain day is recognized by the first 4 hours' data using the SVM-PR method. All 13 solar features are used in the SVM-PR model. Fig. 5 shows sky images and their corresponding $nRBR$ distributions in three clusters. Though the clusters are not meteorologically defined, weather features such as cloud cover and irradiance play critical roles in the clustering and PR.

Three metrics are used to evaluate the PR results, which are sensitivity (S_{tv}), precision (P_{cs}), and accuracy (A_{cc}). S_{tv} is the proportion of labels that are correctly recognized; P_{cs} is the proportion of recognized labels of a cluster that are correct; and A_{cc} is the proportion of the total number of correct recognition. These three metrics are defined as [21]:

$$S_{tv} = \frac{pr_{kk}}{\sum_{k'=1}^{K_{opt}} pr_{kk'}} \tag{24}$$

$$P_{cs} = \frac{pr_{kk}}{\sum_{k'=1}^{K_{opt}} pr_{k'k}} \tag{25}$$

$$A_{cc} = \frac{pr_{kk}}{\sum_{k=1}^{K_{opt}} \sum_{k'=1}^{K_{opt}} pr_{kk'}} \tag{26}$$

TABLE II
PATTERN RECOGNITION RESULTS AND EVALUATION

Result/evaluation		Actual cluster (k) A	B	C
Recognized cluster (k')	A	36	2	0
	B	3	31	8
	C	1	3	11
PR metrics [%]	S_{tv}	90.0	86.1	57.9
	P_{cs}	94.7	73.8	73.3
	A_{cc}		**82.1**	

where $pr_{kk'}$ represents the objects that belong to cluster k and are recognized to cluster k' (k and k' can be identical). The PR results and performance evaluation are listed in Table II. Compared to clusters B ($S_{tv} = 86.1\%$) and C ($S_{tv} = 57.9\%$), objects in cluster A ($S_{tv} = 90.0\%$) are recognized more precisely. Most mistakes are made by categorizing the objects into cluster C ($P_{cs} = 73.3\%$). By only using the first four hours' data, the overall accuracy is 82.1%, which is a significant improvement compared to the direct classification method (A_{cc} of the direct classification method using the first four hours' data is only 13%; to achieve more than 80% A_{cc}, the direct classification method needs more than 11 hours' data, as shown in Fig. 1).

D. Forecasting Results

In the developed UC-M3 method, once a cluster is recognized by SVM-PR, the best-performing M3 model is selected as the forecasting engine for that specific day (the combination of the best-performing M3 models in all clusters is denoted as C_{opt}). Benchmarks include AIO models and SAML models for all clusters in the 4 groups.

1) Forecasting Accuracy Assessment: Two commonly used error metrics are used to evaluate forecasting results, which are normalized mean absolute error ($nMAE$) and normalized root mean square error ($nRMSE$) [11]. The forecasting errors of UC-M3, UC-SAML, AIO-M3, and AIO-SAML groups are summarized in Table III. The best UC-M3 (the upper part of

TABLE III
OVERALL FORECASTING EVALUATION

Group	Model	M3		SAML	
		$nMAE$	$nRMSE$	$nMAE$	$nRMSE$
UC	C_{opt}	***4.79***	***7.94***	***6.37***	***9.74***
	ANN_1	5.83	9.19	7.74	11.38
	ANN_2	5.82	9.14	7.53	11.12
	ANN_3	5.86	9.19	7.72	11.42
	ANN_4	5.71	9.11	8.50	13.73
	SVR_1	6.86	10.53	8.19	12.15
	SVR_2	5.70	9.66	**6.88**	**10.03**
	SVR_3	5.45	**8.50**	7.91	11.28
	GBM_1	5.76	8.87	7.38	10.83
	GBM_2	5.72	8.87	7.39	10.85
	GBM_3	**5.44**	8.99	7.30	11.17
	RF	5.84	9.41	7.21	10.87
AIO	ANN_1	8.04	11.58	10.20	14.63
	ANN_2	7.61	10.97	10.18	14.42
	ANN_3	7.61	11.00	10.25	14.64
	ANN_4	7.74	11.51	11.55	16.37
	SVM_1	7.44	11.47	10.51	14.08
	SVM_2	**6.40**	**9.73**	8.39	11.46
	SVM_3	7.52	10.64	8.59	11.49
	GBM_1	7.29	10.98	8.53	11.78
	GBM_2	7.21	10.89	8.48	11.75
	GBM_3	7.79	11.86	9.49	12.91
	RF	7.83	12.20	9.40	13.56
	P	7.91	11.33	**7.91**	**11.33**

Note: The units of all the evaluation metrics are %. The footnotes of models indicate kernel index of the same ML algorithm. 'P' represents the 1HA persistence of cloudiness method. The best model in all the four groups based on each evaluation metric is highlighted in green, and the best model in each group based on each metric is highlighted in bold.

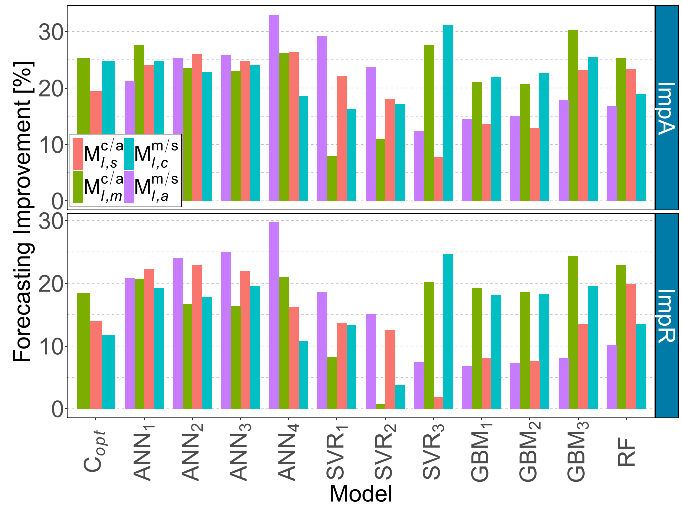


Fig. 6. Improvements of UC-based forecasting over AIO-based forecasting and M3 forecasting over SAML forecasting.

Table III) and UC-SAML models are two C_{opt} models (as highlighted in bold italics). If only a single algorithm is allowed in the M3 second-layer or in a SAML model for different clusters in UC-based forecasting (excluding the C_{opt}), SVR_3 is the best UC-M3 model, while SVM_2 and GBM_3 are the two best UC-SAML models (as highlighted in bold). In the two AIO forecasting groups (the lower part of Table III), SVR_2 and 1HA persistence of cloudiness model (P) outperform other M3 models and SAML models, respectively. Compared to other models in the four groups, UC-M3 based C_{opt} model presents the smallest forecasting $nMAE$ and $nRMSE$ values (as highlighted in green).

2) Superiorities of UC-Based and M3 Forecasting: The superiority of a model over another model can be validated by its forecasting error reduction. Thus, $nMAE$ improvement ($ImpA$) and $nRMSE$ improvement ($ImpR$) are selected in this paper to perform comparisons between UC-based/AIO-based forecasting and M3/SAML forecasting. The ($ImpA$) and ($ImpR$) metrics are defined as:

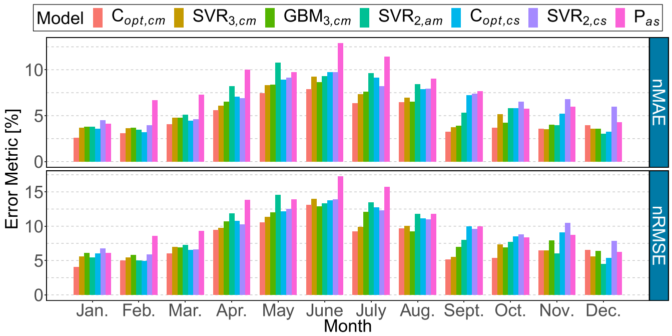
$$ImpA^{j/k} \equiv \frac{nMAE_{M_{l,ij}} - nMAE_{M_{l,ik}}}{nMAE_{M_{l,ij}}} \quad (27)$$

$$ImpR^{j/k} \equiv \frac{nRMSE_{M_{l,ij}} - nRMSE_{M_{l,ik}}}{nRMSE_{M_{l,ij}}} \quad (28)$$

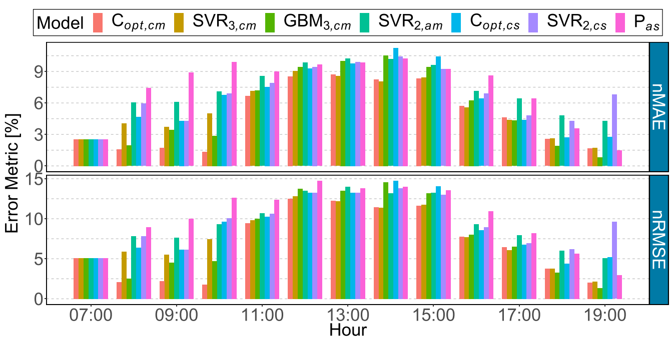
where M is the model name, and l is the kernel index. i, j, k are group indices, which could be c (UC-based group), a (AIO group), m (M3 forecasting group), or s (SAML forecasting group). $ImpA^{j/k}$ and $ImpR^{j/k}$, respectively, are the $nMAE$ and $nRMSE$ improvements of a model in group j compared to the same model in group k.

In this paper, the superiority of the developed UC-M3 solar forecasting method is validated by exploring the effectiveness of both UC-based forecasting and M3-based forecasting. Hence, four comparison counterparts are set, which are UC-SAML/AIO-SAML based forecasting ($M^{c/a}_{l,s}$), UC-M3/AIO-M3 based forecasting ($M^{c/a}_{l,m}$), UC-M3/UC-SAML based forecasting ($M^{m/s}_{l,c}$), and M3-AIO/SAML-AIO based forecasting ($M^{m/s}_{l,a}$). Fig. 6 visualizes the above four comparison counterparts, from which several findings are observed. First, both UC and M3 improve the short-term solar forecasting, since all the $ImpA$ and $ImpR$ values are positive. Second, in the same comparison group, the improvements of different models vary distinctively. For example, the $ImpA$ in $M^{c/a}_{l,m}$ comparison group ranges from **7.89%** ($SVR_{1,m}$) to **30.25%** ($GBM_{3,m}$). Third, the same model achieves different degrees of improvements when combined with different forecasting strategies. For instance, the UC-M3 SVR_2 ($SVR_{2,cm}$) model shows only **0.72%** $ImpR$ compared to AIO-M3 SVR_2 ($SVR_{2,am}$). However, it reduces **15.13%** $nRMSE$ by using the AIO-M3 SVR_2 model ($SVR_{2,am}$) compared with using the AIO-SAML SVR_2 model ($SVR_{2,as}$). The average $ImpA^{c/a}$ and $ImpR^{c/a}$ are **21.04%** and **15.51%**, respectively; and the average $ImpA^{m/s}$ and $ImpR^{m/s}$ are **21.63%** and **16.36%**, respectively. Therefore, it can be concluded that both UC and M3 have improved the short-term GHI forecasting accuracy significantly.

3) Calendar and Weather Effects: It is reported in the literature that the forecasting accuracy of power time series, such as solar and load, is influenced by calendar effects [36] and weather effects [9]. To further explore the calendar and weather effects on the developed method, the best model(s) in each group is(are) picked out to make comparisons, which are C_{opt}, SVR_3, and GBM_3 in the UC-M3 group ($\{C_{opt,cm}, SVR_{3,cm}, GBM_{3,cm}\} \in M_{l,cm}$), C_{opt} and SVR_2 in the UC-SAML group ($\{C_{opt,cs}, SVR_{2,cs}\} \in M_{l,cs}$), SVR_2 in the

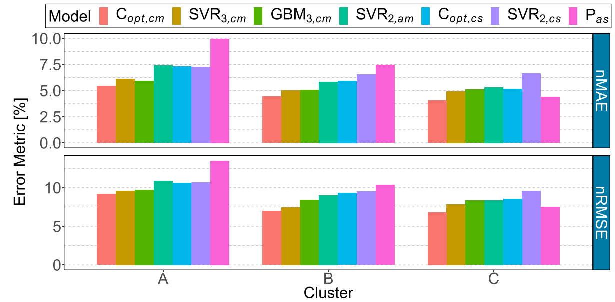


(a) Forecasting errors by month of the year

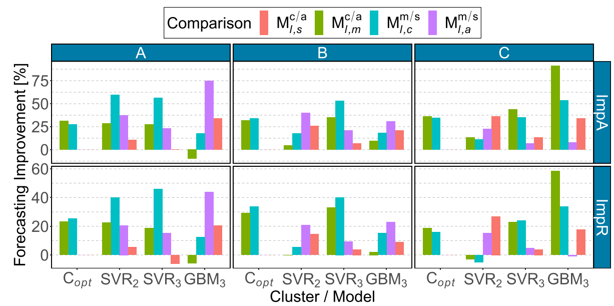


(b) Forecasting errors by hour of the day

Fig. 7. Calendar effects on forecasting errors.



(a) Forecasting errors by cluster



(b) Forecasting improvements by cluster

Fig. 8. Weather effects on forecasting errors.

AIO-M3 group ($\text{SVR}_{2,am} \in M_{l,am}$), and 1HA persistence of cloudiness method in the AIO-SAML group ($\text{P}_{as} \in M_{l,as}$).

Fig. 7 presents forecasting errors of the selected 7 models with respect to calendar units (i.e., month of the year and hour of the day). It is observed that forecasting errors show evident daily and yearly patterns due to calendar effects. The forecasting errors are larger in months or hours that have larger GHI values, such as May–Aug. or 11:00–15:00. It is also found that the $M_{l,cm}$ and $M_{l,am}$ models show superior performance than those of their counterpart groups (i.e., $M_{l,cs}$ and $M_{l,as}$) in most months and hours. Similarly, the $M_{l,cm}$ and $M_{l,cs}$ models generate smaller forecasting errors than the counterparts in $M_{l,am}$ and $M_{l,as}$ in most months and hours, respectively. Compared to other 6 models, $\text{C}_{opt,cm}$ presents better forecasting accuracy in most cases though forecasting error patterns vary due to calendar effects.

Another way to compare forecasting performance of the developed method is to consider weather conditions. Gigoni *et al.* [9] evaluated weather effects based on CSI conditions. In this paper, the weather effects on solar forecasting are explored by comparing the best model(s) in each group directly based on the 3 clusters. Fig. 8 presents forecasting errors and improvements of several best models by cluster. It is observed that models generate smaller errors in cluster C days than in cluster A and cluster B days. This is because cluster A and cluster B have time series with larger GHI values, which lead to larger forecasting variations. Fig. 8(a) also shows that the developed UC-M3 model $\text{C}_{opt,cm}$ outperforms other models consistently. It is found from Fig. 8(b) that both UC and M3 improve the forecasting accuracy significantly in most cases (i.e., cluster/models). For example, the M3 model with GBM_3 as the blending algorithm in the second layer improves cluster C forecasting accuracy by more than **80%** and **50%** based on $ImpA$ and $ImpR$, respectively. In some cases, such as comparing $\text{SVR}^{c/a}_{2,m}$ and $\text{SVR}^{m/s}_{2,c}$, the accuracy of SVR_2 deteriorates by adopting UC-M3 in cluster C. Nevertheless, the SVR_2 model is significantly improved by UC-M3 in clusters A and B forecasting, which compensates for the deterioration in cluter C. Overall, both UC and M3 have improved solar forecasting in each cluster significantly.

V. Conclusion and Future Work

This paper developed an unsupervised clustering and Machine Learning based Multi-Model (UC-M3) framework to perform short-term global horizontal irradiance (GHI) forecasting. An Optimized Cross-validated ClUsteRing (OCCUR) method was developed to determine the optimal number of clusters and generate the best daily GHI time series clustering. Then, support vector machine pattern recognition (SVM-PR) was utilized to recognize the cluster label of a forecasting day, only using the first four hours' solar data (including sky images, GHI features, and weather information). Finally, UC-M3 forecasting was carried out by choosing the best-performing M3 model for each clustered forecasting subtask. Case studies based on 1-year of solar data showed that:

1) The OCCUR method successfully clustered daily GHI time series by using different cross-validated unsupervised clustering methods.
2) SVM-PR recognized daily labels of the data with an overall accuracy of 82.1% by using limited data within a day (i.e., four hours' data in the case study), which made it possible to perform UC-based forecasting.

3) The UC-M3 forecasting method significantly improved the short-term GHI forecasting accuracy, as illustrated by the effectiveness of both UC (average **21.04%** $ImpA$ and **15.51%** $ImpR$) and M3 methods (average **21.63%** $ImpA$ and **16.36%** $ImpR$).
4) The calendar and weather effects analysis indicated the robust and consistent improvements of the developed UC-M3 method.

Future work will focus on utilizing deep learning algorithms in clustering, pattern recognition, and forecasting stages for longer-term solar forecasting.

References

[1] N. Tanaka, "Technology roadmap — Solar photovoltaic energy," *Photovoltaic Power Syst. Programme*, Int. Energy Agency, Paris, France, 2010. [Online]. Available: http://www.iea.org

[2] J. L. Sawin *et al.*, "Renewables 2017-Global status report," Tech. Rep. REN21, Paris, France, 2017.

[3] A. Shakya *et al.*, "Solar irradiance forecasting in remote microgrids using markov switching model," *IEEE Trans. Sustain. Energy*, vol. 8, no. 3, pp. 895–905, Jul. 2017.

[4] Y. Zhang, M. Beaudin, R. Taheri, H. Zareipour, and D. Wood, "Day-ahead power output forecasting for small-scale solar photovoltaic electricity generators," *IEEE Trans. Smart Grid*, vol. 6, no. 5, pp. 2253–2262, Sep. 2015.

[5] C. Voyant *et al.*, "Machine learning methods for solar radiation forecasting: A review," *Renew. Energy*, vol. 105, pp. 569–582, 2017.

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

[7] M. Q. Raza, M. Nadarajah, and C. Ekanayake, "On recent advances in PV output power forecast," *Sol. Energy*, vol. 136, pp. 125–144, 2016.

[8] C. Feng and J. Zhang, "Reinforcement learning based dynamic model selection for short-term load forecasting," 2018, arXiv preprint arXiv:1811.01846.

[9] L. Gigoni *et al.*, "Day-ahead hourly forecasting of power generation from photovoltaic plants," *IEEE Trans. Sustain. Energy*, vol. 9, no. 2, pp. 831–842, Apr. 2018.

[10] C. Feng and J. Zhang, "Hourly-similarity based solar forecasting using multi-model machine learning blending," in *Proc. IEEE Power Energy Soc. General Meeting*, 2018, pp. 1–5.

[11] C. Feng *et al.*, "Short-term global horizontal irradiance forecasting based on sky imaging and pattern recognition," in *Proc. IEEE Power Energy Soc. General Meeting*, 2017, pp. 1–5.

[12] H. S. Jang, K. Y. Bae, H.-S. Park, and D. K. Sung, "Solar power prediction based on satellite images and support vector machine," *IEEE Trans. Sustain. Energy*, vol. 7, no. 3, pp. 1255–1263, Jul. 2016.

[13] X. G. Agoua, R. Girard, and G. Kariniotakis, "Short-term spatio-temporal forecasting of photovoltaic power production," *IEEE Trans. Sustain. Energy*, vol. 9, no. 2, pp. 538–546, Apr. 2017.

[14] J. R. Andrade and R. J. Bessa, "Improving renewable energy forecasting with a grid of numerical weather predictions," *IEEE Trans. Sustain. Energy*, vol. 8, no. 4, pp. 1571–1580, Oct. 2017.

[15] H.-T. Yang, C.-M. Huang, Y.-C. Huang, and Y.-S. Pai, "A weather-based hybrid method for 1-day ahead hourly forecasting of PV power output," *IEEE Trans. Sustain. Energy*, vol. 5, no. 3, pp. 917–926, Jul. 2014.

[16] K. Y. Bae, H. S. Jang, and D. K. Sung, "Hourly solar irradiance prediction based on support vector machine and its error analysis," *IEEE Trans. Power Syst.*, vol. 32, no. 2, pp. 935–945, Mar. 2017.

[17] M. J. Sanjari and H. Gooi, "Probabilistic forecast of PV power generation based on higher order Markov chain," *IEEE Trans. Power Syst.*, vol. 32, no. 4, pp. 2942–2952, Jul. 2017.

[18] M. Pérez-Ortiz, S. Jiménez-Fernández, P. A. Gutiérrez, E. Alexandre, C. Hervás-Martínez, and S. Salcedo-Sanz, "A review of classification problems and algorithms in renewable energy applications," *Energies*, vol. 9, no. 8, 2016, Art. no. 607.

[19] J. Wu and C. K. Chan, "Prediction of hourly solar radiation with multi-model framework," *Energy Convers. Manage.*, vol. 76, pp. 347–355, 2013.

[20] M. Ding, L. Wang, and R. Bi, "An ANN-based approach for forecasting the power output of photovoltaic system," *Procedia Environmental Sci.*, vol. 11, pp. 1308–1315, 2011.

[21] F. Wang, Z. Zhen, Z. Mi, H. Sun, S. Su, and G. Yang, "Solar irradiance feature extraction and support vector machines based weather status pattern recognition model for short-term photovoltaic power forecasting," *Energy Buildings*, vol. 86, pp. 427–438, 2015.

[22] C. Feng, M. Cui, B.-M. Hodge, and J. Zhang, "A data-driven multi-model methodology with deep feature selection for short-term wind forecasting," *Appl. Energy*, vol. 190, pp. 1245–1257, 2017.

[23] F. L. Quilumba, W.-J. Lee, H. Huang, D. Y. Wang, and R. L. Szabados, "Using smart meter data to improve the accuracy of intraday load forecasting considering customer behavior similarities," *IEEE Trans. Smart Grid*, vol. 6, no. 2, pp. 911–918, Mar. 2015.

[24] K. Mets, F. Depuydt, and C. Develder, "Two-stage load pattern clustering using fast wavelet transformation," *IEEE Trans. Smart Grid*, vol. 7, no. 5, pp. 2250–2259, Sep. 2016.

[25] K. Zhang, H. Zhu, and S. Guo, "Dependency analysis and improved parameter estimation for dynamic composite load modeling," *IEEE Trans. Power Syst.*, vol. 32, no. 4, pp. 3287–3297, Jul. 2017.

[26] Y. Liu, R. Sioshansi, and A. J. Conejo, "Hierarchical clustering to find representative operating periods for capacity-expansion modeling," *IEEE Trans. Power Syst.*, vol. 33, no. 3, pp. 3029–3039, May 2018.

[27] O. P. Dahal, S. M. Brahma, and H. Cao, "Comprehensive clustering of disturbance events recorded by phasor measurement units," *IEEE Trans. Power Del.*, vol. 29, no. 3, pp. 1390–1397, Jun. 2014.

[28] C. C. Aggarwal and C. K. Reddy, *Data Clustering: Algorithms and Applications*. Boca Raton, FL, USA: CRC Press, 2013.

[29] C. Ding and X. He, "Cluster merging and splitting in hierarchical clustering algorithms," in *Proc. IEEE Int. Conf. Data Mining*, 2002, pp. 139–146.

[30] J. Handl, J. Knowles, and D. B. Kell, "Computational cluster validation in post-genomic data analysis," *Bioinformatics*, vol. 21, no. 15, pp. 3201–3212, 2005.

[31] S. Aghabozorgi, A. S. Shirkhorshidi, and T. Y. Wah, "Time-series clustering—A decade review," *Inf. Syst.*, vol. 53, pp. 16–38, 2015.

[32] G. Brock, V. Pihui, S. Datta, and S. Datta, "clvalid, an R package for cluster validation," *J. Statist. Softw.*, vol. 25, pp. 1–24, 2008.

[33] A. A. Munshi and A.-R. M. Yasser, "Photovoltaic power pattern clustering based on conventional and swarm clustering methods," *Sol. Energy*, vol. 124, pp. 39–56, 2016.

[34] C. Feng and J. Zhang, "Wind power and ramp forecasting for grid integration," in *Advanced Wind Turbine Technology*. New York, NY, USA: Springer, 2018, pp. 299–315.

[35] C. Feng, M. Sun, M. Cui, E. K. Chartan, B.-M. Hodge, and J. Zhang, "Characterizing forecastability of wind sites in the United States," *Renew. Energy*, doi: 10.1016/j.renene.2018.08.085.

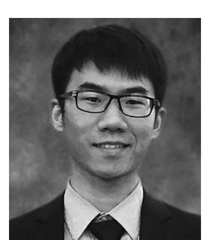
[36] C. Feng and J. Zhang, "Short-term load forecasting with different aggregation strategies," in *Proc. Int. Design Eng. Tech. Conf. Comput. Inf. Eng. Conf.*, Aug. 26–29, 2018, Paper no. DETC2018-86084.

[37] P. Ineichen and R. Perez, "A new airmass independent formulation for the linke turbidity coefficient," *Sol. Energy*, vol. 73, no. 3, pp. 151–157, 2002.

[38] C. Feng, E. K. Chartan, B.-M. Hodge, and J. Zhang, "Characterizing time series data diversity for wind forecasting," in *Proc. 4th IEEE/ACM Int. Conf. Big Data Comput., Appl. Technol.*, 2017, pp. 113–119.

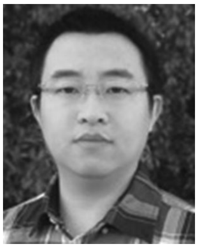
[39] T. Hong, P. Pinson, and S. Fan, "Global energy forecasting competition 2012," *Int. J. Forecasting*, vol. 30, no. 2, pp. 357–363, 2014.

[40] T. Hong, P. Pinson, S. Fan, H. Zareipour, A. Troccoli, and R. J. Hyndman, "Probabilistic energy forecasting: Global energy forecasting competition 2014 and beyond," *Int. J. Forecasting*, vol. 32, no. 3, pp. 896–913, 2016.



Cong Feng (S'17) received the B.S. degree in power engineering from Wuhan University, Wuhan, China, in 2014, and the M.S. degree in mechanical engineering from the University of Texas at Dallas, Richardson, TX, USA, in 2017. He is currently working toward the Ph.D degree with the Department of Mechanical Engineering, University of Texas at Dallas.

His research interests include machine/deep learning, power system big data analytics, and power system time series forecasting.



Mingjian Cui (S'12–M'16–SM'18) received the B.S. and Ph.D. degrees from Wuhan University, Wuhan, Hubei, China, all in electrical engineering and automation, in 2010 and 2015, respectively.

He is currently a Postdoctoral Research Associate with Southern Methodist University, Dallas, TX, USA. He was a Postdoctoral Research Associate with the University of Texas at Dallas, Richardson, TX, USA, in 2016 and 2017. He was also a Visiting Scholar from 2014 to 2015 with the Transmission and Grid Integration Group, National Renewable Energy Laboratory, Golden, CO, USA. He has authored/coauthored more than 50 peer-reviewed publications. His research interests include power system operation, wind and solar forecasts, machine learning, data analytics, and statistics. He serves as an Associate Editor for the journal of *IET Smart Grid*.

Bri-Mathias Hodge (M'10–SM'17) received the B.S. degree in chemical engineering from Carnegie Mellon University, Pittsburgh, PA, USA, in 2004, the M.S. degree from Process Design and Systems Engineering Laboratory, Åbo Akademi, Turku, Finland, in 2005, and the Ph.D. degree in chemical engineering from Purdue University, West Lafayette, IN, USA, in 2010.

He is currently an Associate Professor with the Department of Electrical, Computer and Energy Engineering, University of Colorado Boulder, Boulder, CO, USA and a Fellow of the Renewable and Sustainable Energy Institute. He is also the Chief Scientist for the Power System Design and Studies Group, National Renewable Energy Laboratory, Golden, CO, USA. His research interests include energy systems modeling, simulation, optimization, and wind and solar power forecasting.

Siyuan Lu received the B.S. degree in physics from Fudan University, Shanghai, China, and the Ph.D. degree in physics from the University of Southern California, Los Angeles, CA, USA, in 2001 and 2006, respectively.

He is currently a Manager and a Research Staff Member leading the Data Intensive Physical Analytics Research Group, IBM T.J. Watson Research Center, Yorktown Heights, NY, USA. His research interests include architectures of big geospatial data services and the union of physics and data-driven approaches for modeling complex systems with applications in renewable energy forecasting, climate forecasting, and remote-sensing based land surveying and environmental monitoring.

Hendrik F. Hamann (M'01) received the Ph.D. degree from the University of Göttingen, Göttingen, Germany, in 1995.

He is currently a Senior Manager and a Distinguished Research Staff Member with IBM T.J. Watson Research Center, Yorktown Heights, NY, USA. In 1999, he joined the IBM T.J. Watson Research Center, where he is leading the Physical Analytics and cognitive IoT program. He has authored and coauthored more than 90 peer-reviewed scientific papers and holds over 90 patents and has over 100 pending patent applications. His current research interest includes sensor networks, sensor-based physical modeling, renewable energy, energy management, precision agriculture, system physics, and big data technologies.

Jie Zhang (M'13–SM'15) received the B.S. and M.S. degrees in mechanical engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2006 and 2008, respectively, and the Ph.D. degree in mechanical engineering from Rensselaer Polytechnic Institute, Troy, NY, USA, in 2012.

He is currently an Assistant Professor with the Department of Mechanical Engineering, University of Texas at Dallas, Dallas, TX, USA. His research interests include multidisciplinary design optimization, complex engineered systems, big data analytics, wind and solar forecasting, renewable integration, and energy systems modeling and simulation.