Gated Convolutional Neural Network for Wind Turbine Blade Icing Detection

Weiwei Tian, Xu Cheng, Fan Shi, Guoyuan Li, Shengyong Chen and Houxiang Zhang

Abstract— **Blade icing detection plays an important role in wind turbine protection and maintenance. Employing well trained deep learning model is a promising method for blade ice detection but needs effective neural networks for sensors data analysis. In this paper, we propose a GRU-gated Convolutional Neural Network (GCNN) to better fuse the information between sensors and temporal information for icing detection. Specifically, with the powerful feature extraction capability, Convolutional Neural Network (CNN) can effectively extract the correlation information of multiple sensors data. Then GRU fuses the temporal information of the feature extracted by CNN to perform the gate of the CNN layer to control the information passed on for icing detection. The proposed method is evaluated on monitoring data generated from 25 wind turbines by two wind farms. The experimental results verify the feasibility and effectiveness of the proposed GCNN.**

I. Introduction

Wind turbine blades are prone to accumulate ice in the environment with low temperatures and high humidity. The icing on the blades will not only affect the power generation performance of the wind turbine, but also cause damage to the wind turbine itself, and may even bring safety problems near the power plant. Therefore, measures for wind turbine blade icing are of great significance.

Nowadays, anti-icing [1] methods and deicing [2] methods are used to deal with blade icing problem. The anti-icing technology and deicing technology are generally combined to mitigate blade icing problem due to the poor anti-icing ability of passive anti-icing technology in extreme weather. However, prompt initiation of the deicing procedure relies on timely and accurate icing detection. Conventional icing detection methods estimate ice acceleration according to physical properties [3] of ice or the changes of machine behavior [4]. But these methods are limited by high costs or are insensitive to small amounts of ice [5].

Data-driven methods based on wind turbine monitoring data, especially deep learning methods, have recently attracted lots of attention for achieving high accuracy [6]. In addition, end-to-end models can be trained by suitable datasets and then implemented on real-time sensors data in a real-time way [7]. As one of the most popular neural network for deep learning, Convolutional Neural Networks (CNN) are widely used to process images [8], language [9], and time-series data due to their powerful feature extraction capability. CNN is also used for wind turbine blade icing detection [10]. Sensors data collected from wind turbines are multivariate time series data. CNN can effectively fuse the correlation between time series and extract more detailed local abstract features. Nevertheless, CNN cannot fuse the long-term temporal information of sensors data which is critical for time series data analysis.

Recurrent neural network (RNN) has received an extensive concern for processing sequence data well. The most commonly used RNN variants are long short-term memory (LSTM) [11] cell and gated recurrent unit (GRU) [12], which are investigated to better overcome vanishing gradients problem by introducing a gate mechanism. Their success in analyzing sequence data is often linked to their capability to capture long-term dependencies. Thus, these RNN variants can be leveraged to capture the long-term temporal information of time series data. Compared with LSTM, GRU does not maintain a cell state and has lesser gates, meaning that GRU has fewer parameters and faster training speed.

The models that combine CNN and RNN can analyze multivariate time series data from feature and temporal aspects, and the inputs of CNN and RNN layers are usually the outputs of each other in addition to the original input data. While this can take both of the detailed local abstractive features and long-term temporal dependency into consideration, the effect of fusing these information are not considered and investigated.

To effectively explore and leverage the deep representations and temporal information of the sensors data, GRU-gated Convolutional Neural Network (GCNN) is proposed for wind turbine blade icing detection, in which a gating mechanism is employed to control the deep information extracted by CNN passed on according to the temporal information of sensors data. Additionally, to verify the effectiveness of the proposed model, a complete framework for wind turbine blade icing detection is investigated including data preprocessing, GCNN module training and real-time icing detection. The contribution of this work can be summarized as follows:

1) We propose a GCNN for wind turbine blade icing detection, which can effectively extract the deep features and time information of multi-sensor data and can be fused through a gating mechanism.
2) Based on the proposed GCNN, a wind turbine blade icing detection framework is proposed and evaluated on sensors data generated from 25 wind turbines by two wind farms.

Weiwei Tian, Fan Shi and Shengyong Chen are with the School of Computer Science and Technology, Tianjin University of Technology, Tianjin, 300384, China.

Xu Cheng, Guoyuan Li, and Houxiang Zhang are with the Department of Ocean Operations and Civil Engineering, Norwegian University of Science and Technology, Aalesund, 6009, Norway.

Xu Cheng is also with the Smart innovation Norway, Halden, 1783, Norway.

The rest of the paper is structured as follows. The overview of the literature on wind turbine blade icing detection is illustrated in Section II. Section III describes the proposed model GCNN. Section IV evaluates the proposed model through comprehensive experiments. Section V summarizes the whole paper.

II. Related Work

Data-driven methods for wind turbine blade icing detection mainly include shallow machine learning methods and deep learning methods. Shallow machine learning approaches such as k-Nearest Neighbors [13], Support vector machine [14] were employed for wind turbine icing failure detection. In addition, generalized linear models, random forests, and artificial neural networks are used in [15] for wind turbine condition monitoring. Nevertheless, the deep learning method has attracted much more attention as it can automatically learn high-level representations of sensors data and achieve superior performance. Cheng et al. proposed a novel CNN model for wind turbine icing detection, with a novel temporal attention module helping to determine the importance of sensors and timesteps [5]. And they further investigated a semi-supervised learning based version for blade icing detection [16]. Yuan et al. introduced a wavelet-based CNN model for blade icing detection, which can obtain better performance than conventional machine learning approaches [10]. Tian et al. combined the discrete wavelet decomposition with a multilevel convolutional recurrent neural network for blade icing detection [17]. However, the further fusion of the high-level representations and the temporal information of sensors data are not considered in these methods, which is critical for time series data analyzing.

III. GRU-gated Convolutional Neural Network for Icing Detection

A. Overview

The proposed framework consists of three components: data preprocessing, the GCNN model, and real-time icing detection, as shown in Fig. 1. The data preprocessing is first conducted to obtain clean data, which is then processed to be suitable for model analysis. Second, the processed sensor data sent to the GCNN is first processed by the CNN module to obtain high-level abstractive features. Then GRU is added to obtain the temporal information of the extracted features. To effectively fuse the deep representations with the temporal information of the sensor data, a gating mechanism is introduced to control the discriminative deep features propagate. Further, the fused information is utilized for icing detection. Finally, the optimal GCNN models trained offline are employed to calculate the icing probability in real-time sensor data for icing estimation.

B. Data preprocessing

1) Missing value processing: Raw data collected from sensors in wind turbines are inevitably subject to missing due to various factors such as unexpected system error or human error. General solutions for missing value include data imputation and data substitution. However, unlike the intuitively interpretable images and natural language data, time series data cannot visually check the completion effect or the error caused after completion treatment, so the missing values are dropped.

2) Data split and labeling: In order to meet the input requirements of the proposed model, the discrete sensor data need to be split into fixed time steps. We leverage sliding window sampling to split the sensor data. The sliding window sampling method is generally used for time series data oversampling of the minority class because some data points may still appear in the later data fragments after sliding window sampling. Although duplicate data points exist in each segment, the characteristics of segments are unique because they are composed of all time points in each fragment. So the sliding window sampling method can be used for splitting all the icing and normal data.

3) Data normalization: Data normalization is essential for data preprocessing. The normalization of data is to scale the data into a specific interval to remove the unit limitation of the data and convert it into dimensionless pure value, which can help the simultaneous analysis of indicators with different units.

C. GRU-gated convolutional neural network

The proposed GCNN consists of three parts: CNN, GRU and a gating mechanism. The detailed introductions of each module are as follows.

1) CNN module: Sensor data of wind turbine is prone to have diverse and complex characteristics as wind turbines generally operate in fickle circumstances. Thus, CNN is leveraged for its competitive feature extract capability to capture the discriminative features of sensor data. In GCNN, there have three basic CNN blocks comprised of a convolutional layer, a batch normalization (BN) layer, and a ReLU layer. BN is employed to accelerate the training process and improve the model generalization. The ReLU layer is added to mitigate model overfitting. In our model, we represent the input of CNN as X, $X \in \mathbb{R}^{t \times c}$, where t represents the timestamp of the input time series fragment, and c denotes the channel of the data. The basic CNN block is defined as:

$$X_i = ReLU(BN(W * X_{i-1} + b)), i = 1, 2, 3. \tag{1}$$

where i is the layer number of CNN, and X_i represents the feature map output by $i - th$ CNN layer. '$*$' represent the convolutional operation. The parameters learned by the layer are defined as W and b. As shown in Fig. 1, there are three CNN blocks in the GCNN. We define the CNN blocks with the filter sizes of 128, 256, 128, and the kernel sizes of 7, 5, 3, respectively.

2) GRU module: GRU was proposed to capture long-time dependencies of series data. In this work, GRU is leveraged to catch the temporal correlations of the data points in time series data generated by wind turbine sensors. The input of GRU is the deep representation of the sensor data extracted by CNN instead of the raw input data of the proposed model. Compared to LSTM, GRU has fewer parameters because it

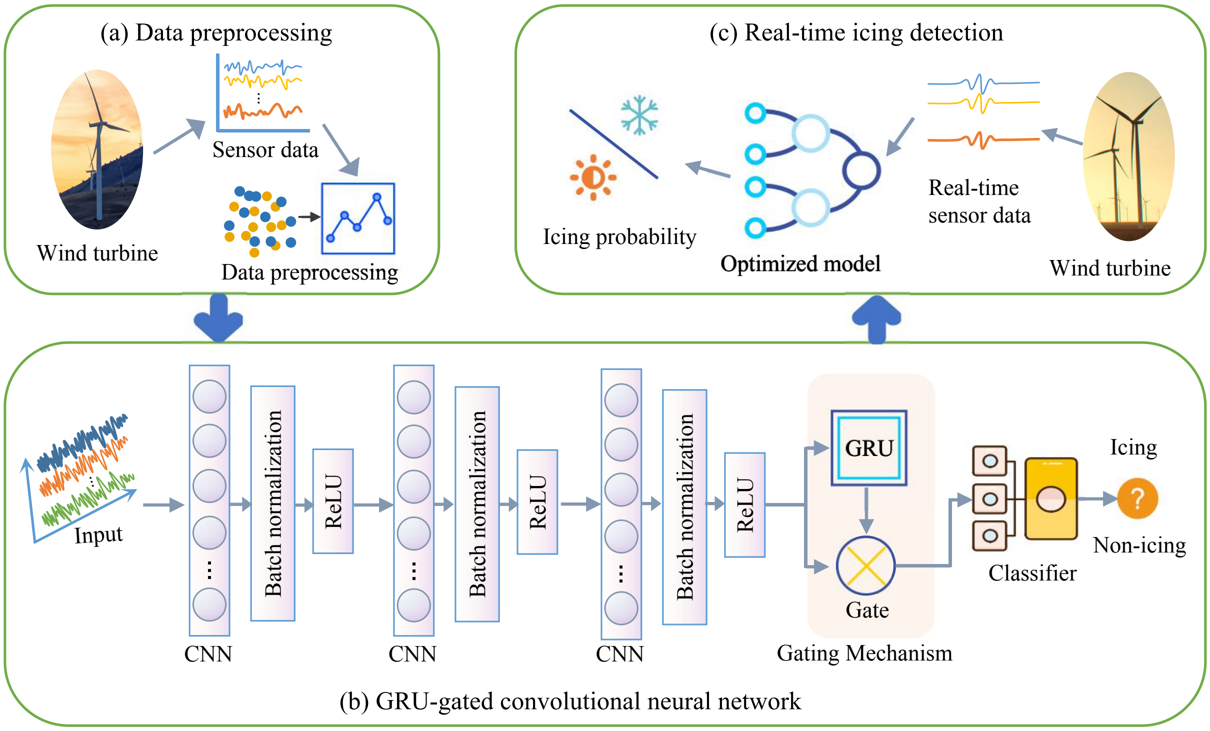


Fig. 1. Overall structure of the proposed method.

only consists of two gates: update date and reset gate, and does not contain a cell state. As shown in Fig. 1, the input of the GRU layer is the feature map of the final CNN layer. We define the input of GRU as $X_3 = [x_0, x_1, x_2, ..., x_t]^T$, where $x \in \mathbb{R}^{1\times c}$. The GRU can be formulated as follows:

$$\begin{aligned} r_t &= \sigma(W_r x_t + U_r h_{t-1}) \\ z_t &= \sigma(W_z x_t + U_z h_{t-1}) \\ h_t &= tanh(W x_t + U_(r_t \otimes h_{t-1})) \\ h_t &= (1 - z_t) h_{t-1} + z_t \hat{h_t} \end{aligned} \tag{2}$$

where r and z are the reset gate and update data, respectively. r can control the reset condition of the value in the GRU unit and z determines the update degree of the activation information. $\hat{h}$ represents the candidate hidden layer, and h represents the hidden layer. W_* and U_* are the weight matrix of GRU. σ represents the logical sigmoid function and $\otimes$ represents the element multiplication operation.

3) Gating mechanism: Gating mechanisms is helpful to control the information passed on in LSTM and GRU cells. Aiming at effectively fuse the features between sensors and temporal dependence of wind turbine monitoring data, we use the temporal information of sensor data as the gate of high-level representations extracted by CNN to control the information propagates for icing detection. So the gating mechanism is added at the end of the proposed neural network. As illustrated in Fig. 1, the gating mechanism fuse the outputs of the CNN module and the GRU layer, so the proposed gate can be defined as: can be computed as:

$$X_{Fused} = X_{CNN} \otimes \sigma(X_{GRU}) \tag{3}$$

where the '$\otimes$' is the element-wise product operation. σ is the sigmoid function.

D. Classifier

The deep features and temporal information fused by the gate unit is then input in the classifier for icing detection. The classifier is used to calculate the icing probability of each time-series fragment. In this work, the feature representations obtained by the GCNN are input to a global average pooling layer, followed by a linear layer combined with the softmax function to calculate the probabilities of icing or not of each sample. The classifier can be computed as follows:

$$y_1 = AveragePooling(X_{Fused}) \tag{4}$$

$$y_2 = Linear(y_1) \tag{5}$$

$$y_3 = Softmax(y_2) = \frac{exp(y_2)}{\sum_{k=1}^{K} exp^{y_2^k}}. \tag{6}$$

where y_* represents the output of each layer. X_{Fused} is the feature map output by gating mechanism.

E. Loss function

The cross entropy (CE) loss function is employed to optimize the proposed model. The CE loss function can be defined as:

$$CE = -y log(\hat{y}) + (1 - y) log(1 - \hat{y}) \tag{7}$$

where y represents the true label of the data, and the positive/negative samples are denoted as 1/0. $\hat{y}$ denotes the probabilities of each class predicted by the model.

F. Real-time icing detection

As shown in Fig. 1 (c), the optimized GCNN model can be utilized to conduct real-time icing detection. Since the input of the GCNN is fixed length data segments, the data detected in real-time should be data streams of the same mode with the training data. The result output by the optimized GCNN model is the probability of blade icing, so it is necessary to set a threshold k to determine whether the blade is frozen. Then the real-time icing detection result is then visualized for real-time observation.

IV. Experiments

A. Experiments settings

1) Dataset: In this work, the sensor data are generated by wind turbines of two wind farms in Shanxi and Henan Provinces of China. The data is collected by Supervisory Control And Data Acquisition system equipped with hundreds of sensors to monitor the wind turbine conditions. The experts identified 16 variables helping to conduct blade icing detection, as shown in Table I. We define the dataset came from Shanxi Province as dataset_1, and the dataset collected from Henan as dataset_2, both of which covered half a month with an interval of 30 seconds for each data point. The raw sensor data inevitably includes noise value, missing values, etc., so we clean up the data before input to the proposed model. The three steps of data preprocessing are described in Section III B.

2) Metrics: We employ F1-score, Area Under Curve (AUC) and Matthews correlation coefficient (MCC) to evaluate the models. The definitions are as follows:

$$F1-score = \frac{2 \times TP}{2 \times TP + FN + FP} \tag{8}$$

$$AUC = \frac{\sum_{i \in positiveClass} rank_i - \frac{M(M+1)}{2}}{M \times N} \tag{9}$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \tag{10}$$

where TP, FP, FN, and TN represent true positive, false positive, false negative, and true negative, respectively. $rank_i$ denotes the number of the $i-th$ sample. M and N represent the number of positive samples and negative samples, respectively.

B. Baseline comparison

We compare the proposed GCNN with four time series baselines on dataset_1 and dataset_2. The time steps of dataset_1 and dataset_2 are 32 and 256, respectively. The details of the baseline networks are as follows.

1) LSTM: LSTM is a commonly used tool for time-series data analyzing. The LSTM employed in this paper for comparison is only one layer with 128 hidden units.

2) GRU: GRU is a simpler variant of LSTM with lesser parameters and faster training speed. The GRU employed for comparison is only one layer with 128 hidden units.

TABLE I

SPECIFICATION OF SENSOR DATA

No.	Variable name	Description
1	wind_speed	Wind speed
2	wind_direction	Wind direction
3	generator_speed	Generator speed
4	power	Active power
5	yaw_position	Yaw position
6	environment_temp	Environment temperature
7	internal_temp	Internal temperature of nacelle
8	pitch1_angle	Angle of pitch 1
9	pitch2_angle	Angle of pitch 2
10	pitch3_angle	Angle of pitch 3
11	pitch1_speed	Speed of pitch 1
12	pitch2_speed	Speed of pitch 2
13	pitch3_speed	Speed of pitch 3
14	pitch1_moto_tmp	Temperature of pitch motor 1
15	pitch2_moto_tmp	Temperature of pitch motor 2
16	pitch3_moto_tmp	Temperature of pitch motor 3

TABLE II

COMPARISON OF TIME SERIES BASELINE ON DATASET_1

	LSTM	GRU	FCN	MLSTM-FCN	GCNN
F1-score	70.9184	69.8734	82.0253	82.3245	**85.0856**
AUC	86.875	87.45	91.925	91.425	**93.275**
MCC	68.3132	66.9091	80.2606	80.5441	**83.5825**

3) FCN: FCN is used for time series classification with competitive performance [18]. The hyper-parameter settings are the same as the original paper.

4) MLSTM-FCN: MLSTM-FCN integrates LSTM and FCN for multivariate time series classification [19].

The comparison results are shown in Table II and Table III. In Table II, the GCNN outperforms all the baseline models and achieves 2.77%, 1.86% and 3.04% higher scores on the three metrics, respectively. As shown in Table III, although the result on AUC is suboptimal, the GCNN gets the highest F1-score of 85.71% and the best MCC of 84.34%. These results suggest that the proposed model can achieve better performance on sensor data collected from different wind farms.

C. Comparison on different model variants

We compare the GCNN with three model variants on dataset_1 and dataset_2 to investigate how can each component of the proposed model obtain the greatest performance.

1) GCNN-V1: GCNN-V1 is the GCNN with LSTM for gating to replace the GRU layer.

2) GCNN-V2: GCNN-V2 is the gating mechanism added following the first CNN layer.

3) GCNN-V3: GCNN-V3 is the gating mechanism added following the second CNN layer.

TABLE III

COMPARISON OF TIME SERIES BASELINE ON DATASET_2

	LSTM	GRU	FCN	MLSTM-FCN	GCNN
F1-score	75.555	75.5556	81.6327	85.1852	**85.7143**
AUC	85.8	83.4	90.4	**94.8**	93
MCC	73.9488	73.9488	79.8571	83.8651	**84.3389**

TABLE IV

COMPARISON OF DIFFERENT MODEL VARIANTS ON DATASET_1

	GCNN-V1	GCNN-V2	GCNN-V3	GCNN
F1-score	79.2079	81.6121	80.2005	**85.0856**
AUC	89.85	90.325	90.5	**93.275**
MCC	77.1106	79.7913	78.2263	**83.5825**

TABLE V

COMPARISON OF DIFFERENT MODEL VARIANTS ON DATASET_2

	GCNN-V1	GCNN-V2	GCNN-V3	GCNN
F1-score	79.2453	70.8333	78.4314	**85.7143**
AUC	90.6	84.4	90.2	**93**
MCC	77.1913	68.3462	76.2457	**84.3389**

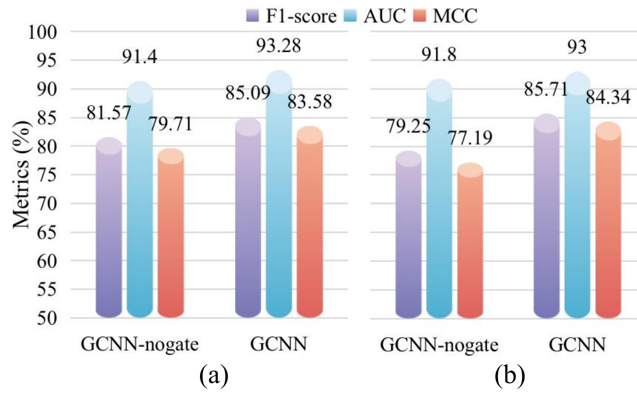


Fig. 2. Ablation experiments on (a) dataset_1, and (b) dataset_2

The results of different model variants comparison on the two datasets are presented in Table IV and V. In table IV, the GCNN can achieve much better performance than the GCNN-V1 on dataset_1, meaning that GRU can do better in gating mechanism than LSTM. Further, the GCNN-V2 and the GCNN-V3 obtain similar scores of all the metrics. However, the results in terms of F1-score, AUC and MCC are 2.95%   5.35% lower than the GCNN. We believe that the reason for this is that, compared to the first two CNN layers, the features extracted by all the three CNN layers are much discriminative for icing detection. The results on dataset_2 shown in Table V are similar to dataset_1. Thus, the rationality and superiority of the GCNN structure are proved.

D. Ablation study

The ablation study is performed to investigate the importance of the proposed gating mechanism. In the ablation study, **GCNN-nogate** is the GCNN without gating mechanism. The results of the ablation study on dataset_1 and dataset_2 are presented in Fig 2. As illustrated in the figure, the GCNN can obtain better performance in terms of all the three metrics on the two datasets. For dataset_1, the GCNN achieves 3.09% and 4.94% higher on average of the three metrics than **GCNN-nogate**, showing a significant improvement with gating mechanism. The results show that the proposed gating mechanism can fuse the feature between sensors and temporal information of the data with competitive capability.

E. Sensitivity analysis

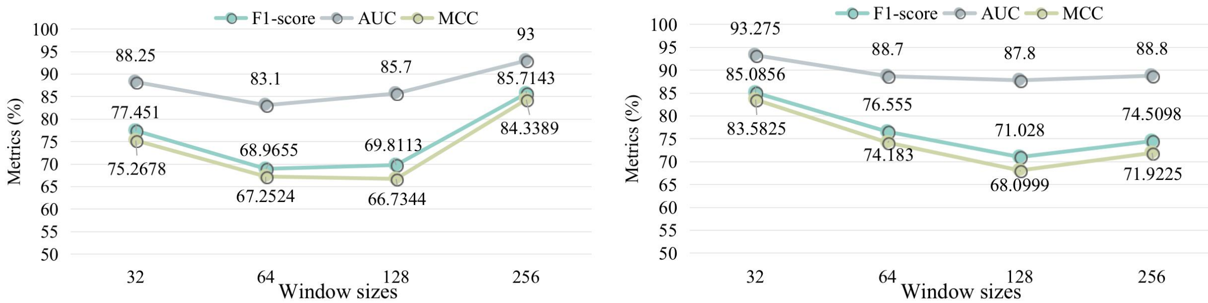
To investigate the influence of window size of data, the performance of the GCNN is compared on two datasets with the segmentation window size is set to 32, 64, 128, 256, respectively. The comparison results on dataset_1 are presented in Fig. 3 (a). It can be seen from the figure, the GCNN with window size of 32 achieves optimal results compared with the other three window sizes. The performance of the GCNN shows a downward trend with the increase of window size. For the comparison on dataset_2, the GCNN can obtain higher scores with 32 window size data than with 64 and 128 window size data. However, when the window size is 256, the performance of the model is significantly improved, which is even better than the performance of the proposed model with window size of 32. The explanation of the different results between the two datasets could be that the operation state of wind turbines might be affected by different geographical environments, resulting in different data characteristics.

F. Real-time detection

The real-time blade icing detection scheme is investigated to provide wind turbine blade icing conditions. After the training phase, the optimal GCNN can be employed to conduct online icing detection. Consistent with the data during the training phase, the available new sensor data streams move forward in sliding window mode to form fixed window size fragments. Then the optimal model output predicted the probability of blade icing on each new sensor data fragment. To further illustrate the results of real-time detection, the visualized probabilities are presented in Fig. 4. As shown in the figure, the blue dotted line indicates the probability threshold of icing and normal. In this paper, the threshold is set to 0.5 for simulation. While in the real scenario, the threshold can be further determined according to the relationship of the real-time detection results and icing conditions. As shown in Fig. 4 (a), the icing area can be successfully detected. While in Fig. 4 (b), there's a false icing alarm in the non-icing area. This is inevitable when wind turbines operate in the real time. So as just mentioned above, the threshold setting for icing determination can be further investigated to reduce the false alarm rate.

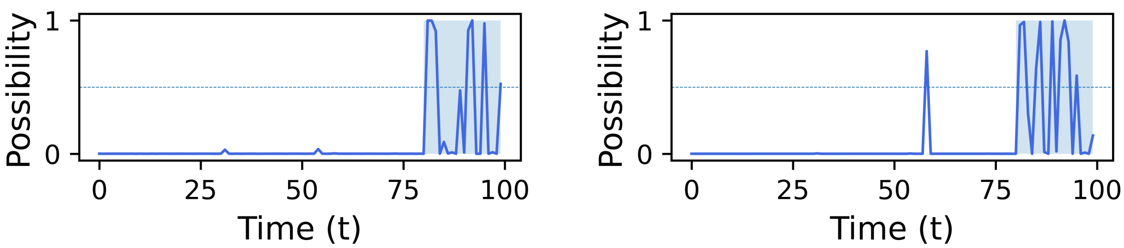
V. CONCLUSION

In this paper, we introduced a novel GRU-gated convolutional neural network for ice accumulation detection on wind turbine blades. The proposed GCNN can fuse CNN with GRU effectively to obtain the discriminative sensor and temporal information of wind turbine monitoring data. The GCNN was evaluated on the datasets collected from 25 wind turbines of two different wind farms. Compared with four state-of-the-art baselines, the GCNN achieve significant improvement in terms of F1-score, AUC and MCC. In addition, the proposed model was compared with three model variants to show the superiority of the GCNN structure. The proposed gating mechanism was further verified by



(a) dataset_1 (b) dataset_2

Fig. 3. Comparison on (a) dataset_1, and (b) dataset_2 with different window sizes.



(a) dataset_1 (b) dataset_2

Fig. 4. Real time detection on these two datasets.

conducting ablation study. The practicability of the proposed model was illustrated by real-time detection.

ACKNOWLEDGMENT

The research is supported by the National Natural Science Foundation of China (62020106004, 92048301).

REFERENCES

[1] N. Dalili, A. Edrisy, and R. Carriveau, "A review of surface engineering issues critical to wind turbine performance," *Renewable & Sustainable Energy Reviews*, vol. 13, no. 2, pp. p.428–438, 2009.

[2] L. Battisti, P. Baggio, and R. Fedrizzi, "Warm-air intermittent de-icing system for wind turbines," *Wind Engineering*, vol. 30, no. 5, pp. 361–374, 2006.

[3] P. Wang, W. Zhou, Y. Bao, and H. Li, "Ice monitoring of a full-scale wind turbine blade using ultrasonic guided waves under varying temperature conditions," *Structural Control & Health Monitoring*, vol. 25, no. 4, pp. e2138.1–e2138.17, 2018.

[4] P. Roberge, J. Lemay, J. Ruel, and A. Bégin-Drolet, "A new atmospheric icing detector based on thermally heated cylindrical probes for wind turbine applications," *Cold Regions Science and Technology*, vol. 148, pp. 131–141, 2018.

[5] X. Cheng, F. Shi, M. Zhao, G. Li, H. Zhang, and S. Chen, "Temporal attention convolutional neural network for estimation of icing probability on wind turbine blades," *IEEE Transactions on Industrial Electronics*, 2021.

[6] H. Ren, B. Xu, Y. Wang, C. Yi, C. Huang, X. Kou, T. Xing, M. Yang, J. Tong, and Q. Zhang, "Time-series anomaly detection service at microsoft," in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp. 3009–3017.

[7] X. Cheng, G. Li, R. Skulstad, S. Chen, H. P. Hildre, and H. Zhang, "A neural-network-based sensitivity analysis approach for data-driven modeling of ship motion," *IEEE Journal of Oceanic Engineering*, vol. 45, no. 2, pp. 451–461, 2019.

[8] C.-F. R. Chen, Q. Fan, and R. Panda, "Crossvit: Cross-attention multi-scale vision transformer for image classification," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 357–366.

[9] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier, "Language modeling with gated convolutional networks," in *International conference on machine learning*. PMLR, 2017, pp. 933–941.

[10] B. Yuan, C. Wang, F. Jiang, M. Long, P. S. Yu, and Y. Liu, "Waveletfcnn: A deep time series classification model for wind turbine blade icing detection," *ArXiv*, vol. abs/1902.05625, 2019.

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

[12] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[13] G. A. Skrimpas, K. Kleani, N. Mijatovic, C. W. Sweeney, B. B. Jensen, and J. Holboell, "Detection of icing on wind turbine blades by means of vibration and power curve analysis," *Wind Energy*, vol. 19, no. 10, pp. 1819–1832, 2016.

[14] Z. Guangfei, T. Wen, and Z. Da, "Ice detection for wind turbine blades based on pso-svm method," *Journal of Physics: Conference Series*, vol. 1087, p. 022036, 09 2018.

[15] P. Song, Z. Yao, and C. Zhao, "Section division and multi-model method for early detection of icing on wind turbine blades," in *2019 34rd Youth Academic Annual Conference of Chinese Association of Automation (YAC)*. IEEE, 2019, pp. 749–754.

[16] X. Cheng, F. Shi, X. Liu, M. Zhao, and S. Chen, "A novel deep class-imbalanced semisupervised model for wind turbine blade icing detection," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[17] W. Tian, X. Cheng, G. Li, F. Shi, S. Chen, and H. Zhang, "A multilevel convolutional recurrent neural network for blade icing detection of wind turbine," *IEEE Sensors Journal*, vol. 21, no. 18, pp. 20 311–20 323, 2021.

[18] Z. Wang, W. Yan, and T. Oates, "Time series classification from scratch with deep neural networks: A strong baseline," in *2017 International Joint Conference on Neural Networks (IJCNN)*, 2017, pp. 1578–1585.

[19] F. Karim, S. Majumdar, H. Darabi, and S. Harford, "Multivariate lstm-fcns for time series classification," *Neural Networks*, pp. 237–245, 2018.