A Fast Abnormal Data Cleaning Algorithm for Performance Evaluation of Wind Turbine

Zhongju Wang, Long Wang, *Member, IEEE*, and Chao Huang, *Member, IEEE*

***Abstract*— A fast wind turbine abnormal data cleaning algorithm via image processing for wind turbine power generation performance measurement and evaluation is proposed in this paper. The proposed method includes two stages, data cleaning and data classification. At the data cleaning stage, pixels of normal data are extracted via image processing based on pixel spatial distribution characteristics of abnormal and normal data in wind power curve (WPC) images. At the data classification stage, wind power data points are classified as normal and abnormal based on the existence of corresponding pixels in the processed WPC image. To accelerate the proposed method, the cleaning operation is executed parallelly using graphics processing units (GPUs) via compute unified device architecture (CUDA). The effectiveness of the proposed method is validated based on real data sets collected from 37 wind turbines of two commercial farms and three types of GPUs are employed to implement the proposed algorithm. The computational results prove the proposed approach has achieved better performance in cleaning abnormal wind power data while the execution time is tremendously reduced. Therefore, the proposed method is available and practical for real wind turbine power generation performance evaluation and monitoring tasks.**

***Index Terms*— Abnormal data cleaning, data-driven approaches, wind power curve (WPC), wind turbines.**

I. Introduction

As an important green energy resource, wind energy has been widely utilized by different countries all over the world. With the increasing installed capacity of wind energy, operations and maintenance (O&M) costs of wind farms have risen correspondingly. Monitoring wind turbine power generation performance is conducive to arranging maintenance plans reasonably, preventing failure occurrences [1], and minimizing O&M cost [2]. To measure the wind turbine power generation performance, accurate wind power curve (WPC) modeling is widely applied.

Manuscript received November 8, 2020; accepted December 7, 2020. Date of publication December 14, 2020; date of current version February 2, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 62002016, in part by the Guangdong Basic and Applied Basic Research Foundation under Grant 2020A1515110431, in part by the Beijing Natural Science Foundation under Grant 9204028, in part by the Beijing Talents Plan under Grant BJSQ2020008, in part by the Scientific and Technological Innovation Foundation of Shunde Graduate School, USTB under Grant BK19BF006 and Grant BK20BF010, and in part by the Interdisciplinary Research Project for Young Teachers of USTB (Fundamental Research Funds for the Central Universities) under Grant FRF-IDRY-19-017. The Associate Editor coordinating the review process was Eduardo Cabal-Yepez. *(Corresponding author: Long Wang.)*

The authors are with the School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing 100083, China, also with the Shunde Graduate School of University of Science and Technology Beijing, Shunde 528300, China, and also with the Beijing Key Laboratory of Knowledge Engineering for Materials Science, Beijing 100083, China (e-mail: long.wang@ieee.org).

This article has supplementary downloadable material available at https://doi.org/10.1109/TIM.2020.3044719, provided by the authors.

Digital Object Identifier 10.1109/TIM.2020.3044719

With the development of information technologies, current wind turbines are generally equipped with data acquisition and monitoring systems. Supervisory control and data acquisition (SCADA) systems that accumulate vast historical data of wind turbines are widely used for various condition monitoring tasks of wind turbines [3], such as fault diagnosis [4] and failure detection [5], [6]. However, wind turbines generally work in environments of large uncertainties so that various abnormal data would be interfused into operational data sets. As a result, these abnormal data seriously influence the average power generation performance evaluation of wind turbines, leading to unreasonable maintenance plans and wasting of resources [7]. Thus, accurate detection of abnormal wind power data benefits wind farm operations.

In the literature, various methods were developed to detect and clean abnormal wind power data. Among them, statistical data diming methods were widely applied. Zheng *et al.* [8] divided wind power data into six categories according to their attribute magnitudes from a statistical perspective. The degree of similarity based on a weighted distance between individual objects was computed and the local outlier factor (LOF) algorithm was applied to identifying the outliers. However, the accuracy of this method cannot be guaranteed using a small number of data points. Wang *et al.* [9] applied the linear mixture self-organizing maps algorithm for wind turbines anomaly detection and fault classification. Yesilbudak [10] used k-means clustering based on squared Euclidean and city-block distance to filter abnormal data according to the Mahalanobis distance thresholds. However, this approach often misclassified stacked outliers as normal data. Zhao *et al.* [11] employed the quartile method to exclude sparse outliers and utilized the density-based clustering method to exclude the stacked outliers. The main limitation of this approach was that several parameters need to be manually tuned for different data sets. Shen *et al.* [12] proposed a method to filter the stacked outliers using the change point grouping algorithm and the quartile algorithm. However, the high computational overhead reduced its capability of handling large data sets.

WPC modeling-based methods are another main type of method for abnormal data cleaning. Based on the spatial distribution characteristics, wind power data points distributed outside the boundary of WPC are considered as outliers. Verma [13] utilized a five-year historical data set to fit the WPC and applied the k-means clustering algorithm to distinguish normal wind power data points and outliers. However, the parameter k needs to be set properly for each wind turbine. Wang *et al.* [14] developed a copula-based joint probability model to fit WPCs and outliers were detected based on the derived joint probability distribution. Ye *et al.* [15]

identified abnormal wind power data points based on the probabilistic WPC and corrected the detected outliers via the spatial correlation characteristics of neighboring wind farms. Considering abnormal data affects the error distribution in WPC modeling, Wang *et al.* [16] proposed two novel asymmetric spline regression models, mixture of asymmetric Gaussian and mixture of asymmetric exponential power, for solving the problem of complex and asymmetric error distribution in WPC modeling. However, these methods required a selected training data set to fit WPC models so that abnormal data cleaning performance was influenced by the selection of the training data set.

Recently, image processing approaches have been employed for wind turbine inspection, such as blade surface crack location [17], blade crack segmentation [18], and icing thickness detection of power transmission lines [19]. Image-based methods were also developed for solving the problem of abnormal wind power data cleaning. As a pioneer study, Long *et al.* [20] extracted the normal data via mathematical morphology operation (MMO) based on WPC images. In the proposed method, the principal part of the WPC image was extracted via MMO, and the mapping relationship between wind power data points and WPC image pixels was established to mark the normal and abnormal data. Su *et al.* [21] proposed an image thresholding method for WPC abnormal data cleaning. The proposed method filtered the abnormal data by thresholding on a gray level feature image. However, these image-based methods generally contain algorithm-specific parameters such as the size of the structuring element and the gray threshold value, and tuning these parameters induces additional computational cost.

In conclusion, existing abnormal data cleaning methods for wind power data have the following limitations: 1) high computational overhead is required; 2) cleaning performance is seriously influenced by the selection of training data; and 3) hyperparameters need to be determined for different data sets.

Considering the above-mentioned drawbacks, a simplified version of spatial continuity-based image processing algorithms [22]–[25] is developed in this paper, and it is employed to rapidly perform abnormal data cleaning for wind turbine power generation performance evaluation. The proposed method includes two stages, data cleaning and data classification. Considering the features of WPCs, the data cleaning stage filters abnormal data based on pixel spatial distribution characteristics of abnormal and normal data in WPC images. The mapping rule based on Long *et al.*'s method [20] is applied in the data classification stage. Then, wind power data points are classified as normal and abnormal based on the existence of corresponding pixels in the processed WPC image. Compared with the MMO algorithm used in Long *et al.*'s work [20], the proposed method has the following advantages:

1) Different from general image processing algorithms, shapes and distribution of wind power data outliers are well considered in the proposed method.
2) No algorithm-specific parameters, such as the structuring element size in MMO, are included in the proposed method.

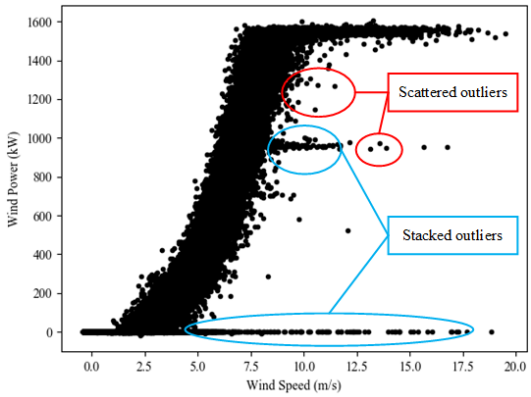

Fig. 1. Abnormal wind power data points.

3) High cleaning efficiency is achieved via simple pixel operations.
4) The parallelism and graphics processing unit (GPU)-based implementation are considered to further reduce the computing time.

Since no algorithm-specific parameters are included in the proposed algorithm, it is more adaptive for different types of wind turbines. Meanwhile, the proposed algorithm does not require any training data, avoiding training data selection. Considering a large amount of data included in the wind power data, the cleaning operation is executed parallelly using GPUs via compute unified device architecture (CUDA), tremendously reducing the computation time of the proposed algorithm. Based on the processed wind power data, WPC models are developed for power generation evaluation.

Due to the time-varying air density and surface roughness of blades [26], WPC shifts over time. A normal WPC region, instead of a fixed curve, is finally obtained via the proposed method and thus the proposed method can reflect the continued uncertainties [27] in wind energy resources.

II. Motivation and Contributions

WPC is used to describe the functional relationship between wind speed and power output. However, outliers of operational data sets induce abnormal curvatures of WPCs, which seriously affects wind turbine power generation performance evaluation. In particular, when outliers take a large percentage of the data set, they are easily misidentified as normal data. According to the spatial distribution of abnormal data points in WPC images, abnormal data, caused by the discrete uncertainties in wind energy, are classified into two categories, scattered outliers and stacked outliers, shown in Fig. 1.

From Fig. 1, it is observable that scattered outliers are randomly distributed in a WPC image. Reasons for scattered outliers include sensor faults, sensor noise, extreme weather conditions, and so on. Differently, stacked outliers are compactly distributed in the horizontal direction and form data clusters. These outliers are generated by unplanned maintenance, wind curtailment, communication failures, and so on.

How to quickly and accurately clean outliers, identify normal and abnormal data clusters, as well as further evaluate the wind turbine power generation performance has attracted wide attention from wind energy academia and industry over the years. Therefore, this paper develops a fast abnormal

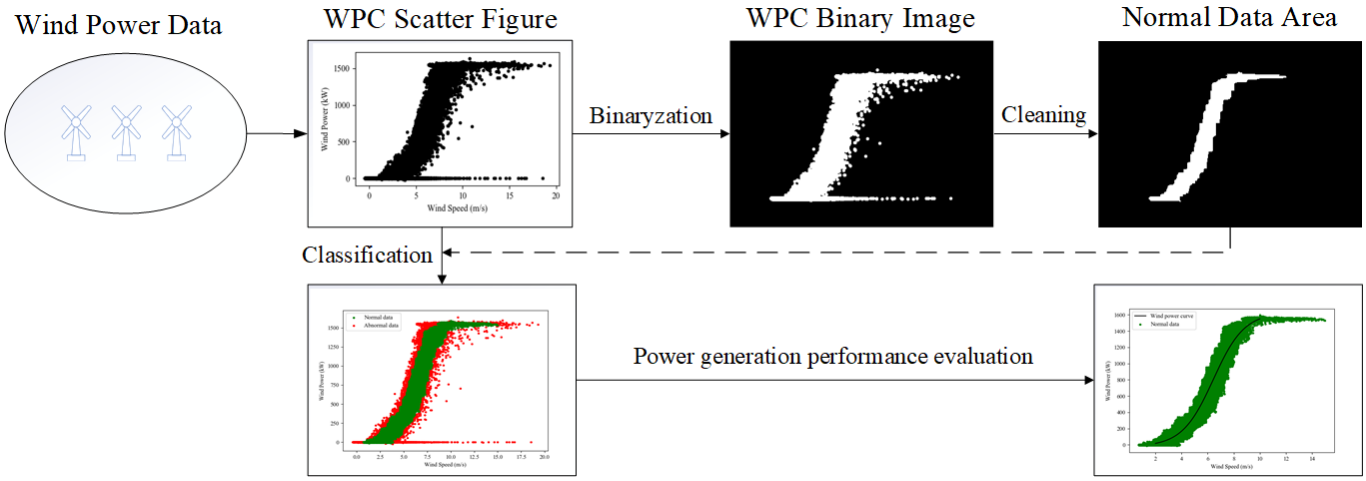

Fig. 2. Schematic of the proposed algorithm.

data cleaning algorithm based on WPC images. The main contributions of this study are summarized as follows:

1) An efficient two-stage data cleaning method based on a simple image processing algorithm is developed to detect and clean abnormal wind power data for WPC measurement.
2) The parallelism of the proposed algorithm is designed, and the GPU is employed to accelerate the proposed algorithm via CUDA.
3) The feasibility and efficiency of the proposed algorithm are validated based on real operational data collected from two commercial wind farms.

III. Proposed Method

In this section, the overall procedures of the proposed method are first introduced. Next, the details of the WPC normal data extracting operation executed on CPU and GPU are described. The main steps of the proposed algorithm are shown in Fig. 2, including:

1) Convert the scatter figure of wind power data to a WPC binary image.
2) Clean abnormal pixels in the WPC binary image using pixel operations.
3) Classify the wind power data point based on the existence of corresponding pixels in the processed WPC image.
4) Evaluate power generation performance via WPC models based on the extracted normal data.

A. WPC Abnormal Data Cleaning on CPU

In WPC binary images, normal wind power data points are normally distributed at the center of WPCs, and abnormal wind power points are distributed around WPCs. Different from abnormal data, normal data have a good coherence in both horizontal and vertical directions in WPC images. Abnormal data points are usually scattered or stacked in the horizontal direction. Based on the spatial distribution difference between the normal and abnormal data, pixels of normal data can be identified via pixel operations incorporating image thresholding [28] and seeded region growing techniques [29].

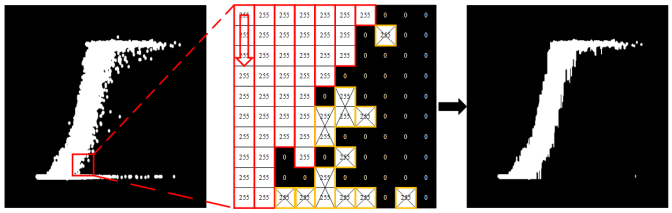

Fig. 3. Vertical cleaning phase.

The cleaning stage includes two phases, vertical cleaning and horizontal cleaning. Stacked outliers are compactly distributed in the horizontal direction in WPC images and they can be cleaned by the vertical cleaning phase effectively. Besides, scattered outliers have little influences on the horizontal coherence of normal data points, and they can be effectively removed via the horizontal cleaning phase. In vertical cleaning, each column of the WPC binary image is traversed. Only the maximum continuous white pixels are retained. Suppose the WPC binary image is $f(x, y)$ denoting the pixel value at position (x, y) and the image size is $m \times n$, the maximum coherence in the vertical direction of the WPC binary image is computed in the following equations:

$$U_{k,y} = \{(s_{k,y}, e_{k,y}) | f(x, y) = 255, 0 < s_{k,y} < x < e_{k,y} < m\} \tag{1}$$

$$d_{\text{best},y} = \max\{e_{1,y} - s_{1,y}, e_{2,y} - s_{2,y}, \ldots, e_{t,y} - s_{t,y}\} \tag{2}$$

$$U_{\text{best},y} = \{(e_{k,y}, s_{k,y}) | e_{k,y} - s_{k,y} = d_{\text{best},y}\} \tag{3}$$

where $U_{k,y}$ represents the kth continuity interval at the yth column, $s_{k,y}$ and $e_{k,y}$ represent the start point and the endpoint of the kth continuity interval at the yth column, $d_{\text{best},y}$ denotes the maximum interval length at the yth column, and $U_{\text{best},y}$ represents the maximum continuity interval at the yth column.

The vertical cleaning phase is depicted in Fig. 3. As shown in Fig. 3, in the vertical cleaning phase, continuous pixels in red boxes are retained and those located in yellow boxes are removed. The processed image $g(x, y)$ after vertical cleaning

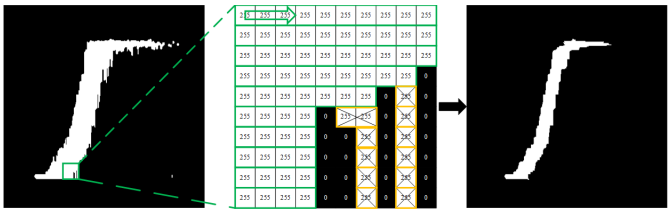

Fig. 4. Horizontal cleaning phase.

is shown in the following equation:

$$g(x, y) = \begin{cases} 255, & \text{if } (x, y) \in U_{\text{best},y} \\ 0, & \text{if } (x, y) \notin U_{\text{best},y}. \end{cases} \tag{4}$$

Similarly, the horizontal cleaning phase is conducted on the results of the vertical cleaning and it is illustrated in Fig. 4. The row-wise interval set $V_{k,i}$ and the best interval V_{best} are obtained correspondingly. Besides, the impacts of the order of two cleaning phases are analyzed in the supplementary materials.

The WPC abnormal data cleaning algorithm on CPU is described as Algorithm 1 in detail.

Algorithm 1 WPC Abnormal Data Cleaning Algorithm on CPU

Input: Pixel value of WPC image $f(x, y)$
Output: Pixel value of the extracted image $f'(x, y)$
Initialize $g(x, y) := 0$, $f'(x, y) := 0$
for j: $=1$ to n **do**
  for i: $=1$ to m **do**
    $U_{k,j} = \{(s_{k,j}, e_{k,j}) | f(i, j) = 255, 0 < s_{k,j} < i < e_{k,j} < m\}$
  end
    $d_{best,j} = \max\{e_{1,j} - s_{1,j}, e_{2,j} - s_{2,j}, \ldots, e_{t,j} - s_{t,j}\}$
  for k: $=1$ to t **do**
    if $e_{k,y} - s_{k,y} = d_{best,y}$ **do**
      $g(s_{k,y} : e_{k,y}, j) := 255$
    end
  end
end
for i: $=1$ to m **do**
  for j: $=1$ to n **do**
    $V_{k,i} = \{(s_{k,i}, e_{k,i}) | g(i, j) = 255, 0 < s_{k,i} < j < e_{k,i} < n\}$
  end
    $d_{best,i} = \max\{e_{1,i} - s_{1,i}, e_{2,i} - s_{2,i}, \ldots, e_{t,i} - s_{t,i}\}$
  for k: $=1$ to t **do**
    if $e_{k,i} - s_{k,i} = d_{best,i}$ **do**
      $f'(i, s_{k,i} : e_{k,i}) := 255$
    end
  end
end

B. Parallel WPC Abnormal Data Cleaning on GPU

To further speed up the computation, the cleaning operation is executed in a parallel way on a GPU using CUDA. CUDA, a parallel computing platform, was introduced by Nvidia in 2007 [30]. It takes advantage of the parallel computing capabilities of GPUs to achieve a significant improvement in computing performance. Furthermore, developers can use C or C++ language to write programs for the CUDA architecture and solve complex problems containing a large number of operations [31]. As a mainstream parallel computing platform, CUDA has strongly promoted the development of computer vision and deep learning algorithms [32]. In the CUDA programming model, CPU and GPU work cooperatively. The CPU is responsible for logical transaction processing and serial computing while the GPU focuses on performing highly threaded parallel processing tasks. Besides, a parallel computing function on GPU is defined as a kernel, and the kernel is executed in blocks. A thread block composed of multiple threads is represented in different dimensional ways, and each block is executed in parallel. A grid, composed of multiple thread blocks, can be organized in the same way. An example structure of a 2-D grid in the CUDA is shown as Fig. 5.

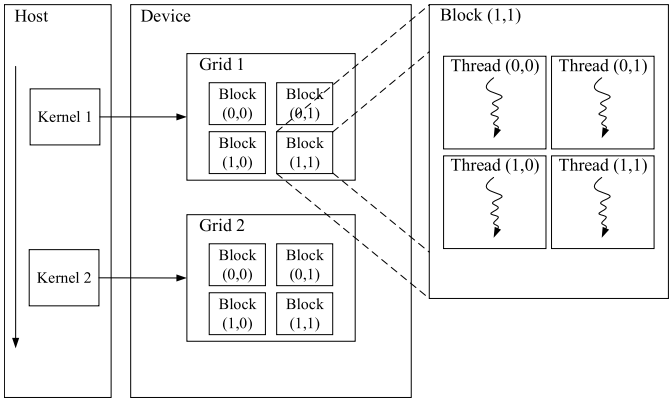

Fig. 5. Threading batching in CUDA.

In this study, CUDA is applied parallelly executing the WPC abnormal data cleaning algorithm on GPU. In the GPU implementation, the vertical cleaning phase is first executed in a parallel way. Next, the horizontal cleaning phase is performed parallelly based on the obtained results of the vertical cleaning phase. Six CUDA kernels are developed correspondingly. Among them, three kernels are used for vertical cleaning and the rest kernels are employed for horizontal cleaning. In this study, the block size and the grid size are set to 1024 and $m \times n/1024$, respectively. All the employed threads execute the same operation parallelly which dramatically improves the efficiency of the data cleaning operation. The parallel WPC abnormal data cleaning algorithm on GPU is illustrated in Algorithm 2.

C. Data Classification

Normal wind power data are obtained after the data cleaning stage while scattered outliers and stacked outliers are removed. At the data classification stage, the original wind power data points corresponding to the removed pixels are marked as abnormal, while those corresponding to pixels of the extracted WPC image are marked as normal.

Inspired by the work of Long *et al.* [20], linear interpolation is used for the mapping between wind power data points and WPC image pixels. Suppose the processed image after vertical

Algorithm 2 Parallel WPC Abnormal Data Cleaning Algorithm on GPU

Input: Pixel value of the WPC binary image $f(x, y)$
Output: Pixel value of the extracted WPC binary image $f'(x, y)$
Set the block size and grid size
Initialize $g(x, y) := 0$, $f'(x, y) := 0$
Map all threads to each pixel of the WPC binary image
for each pixel in $f(x, y)$ **do in parallel**
  Check its neighbors in the vertical direction
  Get $U_{k,y}$
end
Get U_{best} via **parallel reduction**
for each pixel in $g(x, y)$ **do in parallel**
  if (x, y) in U_{best} **do**
   $g(x, y) := 255$
  end
end
for each pixel in $g(x, y)$ **do in parallel**
  Check its neighbors in the horizontal direction
  Get $V_{k,i}$
end
Get V_{best} via **parallel reduction**
for each pixel in $f'(x, y)$ **do in parallel**
  if (x, y) in V_{best} **do**
   $f'(x, y) := 255$
  end
end

cleaning and horizontal cleaning is $f'(x_i, y_i)$, wind power data is classified according to the following equation:

$$\text{Label}(v_i, P_i) = \begin{cases} \text{Normal}, & \text{if } f'(x_i, y_i) = 255 \\ \text{Abnormal}, & \text{if } f'(x_i, y_i) = 0. \end{cases} \tag{5}$$

IV. Benchmarking Methods

To assess the performance of the proposed method, it is benchmarked against MMO [33] and LOF [34] algorithms in this study.

A. Mathematical Morphology Operation

MMO is a mathematical method for deriving the shape and structure of objects in digital images. It has wide applications including edge detection [35] and noise removal [36]. Since the MMO is utilized in Long *et al.*'s work [20] to extract normal data regions of the WPC binary image, it is employed as a benchmarking method in this study. The MMO has two basic operations, expansion and dilation. The expansion operation can maintain basic shape features of objects, and it is widely used for outer noise removal. The dilation operation can remove irrelevant structures of objects, and it is applied to removing the inner noise of the object.

Given a binary image, A, with the structuring element, B, the expansion and dilation operations are described as (6) and (7), separately

$$A \ominus B = \{x, y | (B)_{xy} \subseteq A\} \tag{6}$$

$$A \oplus B = \{x, y | (B)_{xy} \cap A \neq \emptyset\} \tag{7}$$

where $\ominus$ and $\oplus$ denote the erosion operation and the dilation operation, respectively. Besides, other MMOs can be generated by combining erosion and dilation operations. The opening operation is expressed as $(A \ominus B) \oplus B$ and the closing operation is expressed as $(A \oplus B) \ominus B$.

B. Local Outlier Factor

LOF algorithm is a density-based anomaly detection algorithm, which is widely used for anomaly detection [37]. It identifies outliers by comparing the density of each data point with its neighborhood points, and it computes a LOF value which is used to reflect the abnormality of a data point. The lower LOF value of a data point is, the more likely this point is to be an outlier. Compared with conventional statistics methods, the LOF algorithm is more intuitive and simpler. It does not require the distribution of data to quantify the degree of the abnormality of each data point. A more detailed description can be found in [37].

Given any positive integer k, the reachable distance from data point p to data point q is computed in the following equation:

$$\text{reach\_dist}_k(p, q) = \max\{k - \text{dis}(q), d(p, q)\} \tag{8}$$

where $k - \text{dis}(q)$ is the distance between data point q and the kth nearest data point near q, and $d(p, q)$ denoted the distance between p and q. For the data point p, the local reachability density is defined as the following equation:

$$lrd_k(p) = \frac{|N_k(p)|}{\sum_{q \in N_k(p)} \text{reach\_dist}_k(p, q)} \tag{9}$$

where $|N_k(p)|$ is the k-nearest neighborhood of p which represents the number of all points within the kth distance of p, and $|N_k(p)| \geq k$. The LOF score is calculated by the following equation:

$$\text{LOF}_k(p) = \frac{\sum_{q \in N_k(p)} \frac{lrd_k(q)}{lrd_k(p)}}{|N_k(p)|}. \tag{10}$$

V. Numerical Study

A. Data Description

To validate the feasibility of the proposed abnormal wind power data cleaning algorithm, wind power data sets collected from Gaojiagou wind farm, Shanxi, China, and Matang wind farm, Jiangsu, China, are analyzed. The specification of wind turbines is presented in Table I. All considered algorithms are implemented on a PC with AMD Ryzen 3 3200g at 3.6 GHz CPU and 16 GB RAM. The program based on Python 3.7 is executed on Windows 10.

TABLE I
WIND TURBINE SPECIFICATION

Parameter	Gaojiagou	Matang
Cut-in Speed (m/s)	2.5	3.0
Rated Speed (m/s)	13.0	11.0
Cut-out Speed (m/s)	25.0	21.0
Rated Power (kW)	1500	1500
No. of Wind Turbines	20	17

Fig. 6. Reference WPC image.

TABLE II
MAIN TYPES OF ERRORS IN THE FAULT LOG

Component	Errors
Rotor Blades	rotating speed error, yaw error, pitch system error
Gearbox	oil temperature error, oil pressure error, vibration error
Generator	rotor error, converter error, temperature error
SCADA system	wind speed sensor fault, communication interruption

B. Performance Assessment

The reference WPC image generated from a normal wind turbine is employed to evaluate the normal data area yielded by the proposed method automatically. Abnormal data of the selected wind turbine is manually removed by a human expert via fault log and the remaining normal data are used to generate the reference WPC image, shown in Fig. 6. The main types of errors in the fault log are listed in Table II.

To assess the abnormal data cleaning accuracy and efficiency, both outlier detection rate, $R(\%)$, and execution time, $T(s)$, are considered. The outlier detection rate is defined as the following equation:

$$R = \frac{N}{N_{\text{refer}}} \times 100\% \tag{11}$$

where N is the number of outliers filtered by the proposed algorithm and N_{refer} is the number of abnormal data obtained via the reference WPC image.

Besides, the speedup ratio of the GPU-based implementation is defined as the following equation:

$$\lambda = \frac{T_{\text{CPU}}}{T_{\text{GPU}}} \tag{12}$$

where λ is the speedup ratio, T_{CPU} and T_{GPU} denote the execution time on CPU and GPU, respectively.

C. Experimental Results and Discussion

The image resolution of WPC images is set to 432×288 to keep consistency with the compared methods and the size of each wind power data point is 2×2. According to Long *et al.*'s study [20], the structuring element size of MMO is set to 5×5. The similarity of data in LOF is calculated based on the weight distance. The distance parameter k is set to 300 and the LOF threshold value is set to 10% of all LOFs. The program is executed 10 times and the average execution time is recorded. The performance of all the considered methods based on data collected from the two wind farms is shown in Tables III and IV. The results of benchmarking methods are obtained from [20]. To further illustrate the performance of the proposed method, the detection results of example wind turbines are graphically presented in Fig. 7.

According to the results presented in Tables III and IV, it is obvious that the proposed method outperforms the MMO and LOF algorithms over data sets of the 37 wind turbines. In particular, the maximum outlier deletion rate of the proposed method is 65.27% (wind turbine M-07) and it is 40.35% higher compared with that of the MMO (24.92% on wind turbine G-18). The worst detection results are yielded via using the LOF algorithm and its highest detection rate is 9.99%. Furthermore, considering the percentage of total method time that the data cleaning process accounts for, it is observed that the amount of data points is larger, the time percentage of the data cleaning process is smaller. The detection rate of the proposed method on different sizes of data sets is analyzed in the supplementary materials. Besides, regarding the efficiency, the proposed method dominates all three methods in terms of the shortest computing time. Therefore, it is more effective and efficient to apply the proposed method for abnormal wind power data cleaning compared with the MMO and LOF algorithms.

Detection results of the considered methods on wind turbine G-18 and M-15 are shown in Fig. 8. It is seen that both MMO and LOF algorithms fail to detect certain outliers while the proposed method obtains the best detection performance. However, when the proposed method is applied to small data sets, outlier holes may exist inside the WPC images. Thus, after two cleaning phases, some normal pixels may be incorrectly removed.

To evaluate the acceleration performance of the GPU-based implementation, three Nvidia GPUs, GTX 1050Ti, GTX 1080Ti, and RTX 2080Ti are employed to run the proposed data cleaning algorithms. The detailed specifications of these three GPUs are presented in Table V, and the average computing time and speedup ratios of the proposed method on three GPUs are recorded in Table VI.

From Table VI, we see that the execution time of the data cleaning operation is tremendously reduced by utilizing the GPU acceleration and at least a $45\times$ speedup ratio is obtained compared with the CPU-based implementation. The RTX 2080Ti GPU offers the best acceleration performance with a nearly $160\times$ speedup ratio. Furthermore, the speedup ratio of RTX 2080Ti GPU is nearly 3.5 times that of GTX 1050Ti.

The GPU-based parallel MMO [38] and LOF [39] algorithms are utilized to further assess the efficiency of the proposed method. The execution time of different GPU-based

TABLE III
COMPUTATIONAL RESULTS OF DIFFERENT METHODS ON GAOJIAGOU WIND FARM

WT	Amount of data	The proposed algorithm						MMO		LOF	
		Data Cleaning		Data Classification		Total					
		T_{CPU} (s)	Time Percentage	T_{CPU} (s)	Time Percentage	R (%)	T_{CPU} (s)	R (%)	T_{CPU} (s)	R (%)	T_{CPU} (s)
G-01	31266	0.44	55%	0.36	45%	33.51	0.80	20.27	2.72	9.93	5.68
G-02	31157	0.45	55%	0.37	45%	39.63	0.82	15.87	2.62	9.95	6.42
G-03	31277	0.47	57%	0.36	43%	56.94	0.83	10.98	2.57	9.96	5.73
G-04	30862	0.44	53%	0.39	47%	29.03	0.83	17.91	2.86	9.94	5.65
G-05	30756	0.44	55%	0.36	45%	25.98	0.80	8.82	2.57	9.95	5.47
G-06	30924	0.45	54%	0.39	46%	34.14	0.84	13.85	2.77	9.92	5.55
G-07	31401	0.45	56%	0.36	44%	43.35	0.81	13.86	2.54	9.96	5.62
G-08	31157	0.45	52%	0.41	48%	31.81	0.86	20.05	2.67	9.93	5.65
G-09	31059	0.45	57%	0.34	43%	61.66	0.79	20.74	2.54	9.94	5.62
G-10	31421	0.44	52%	0.40	48%	48.06	0.84	10.26	2.53	9.96	5.63
G-11	31450	0.45	56%	0.36	44%	31.91	0.81	15.81	2.53	9.94	5.58
G-12	31343	0.45	52%	0.41	48%	40.66	0.86	12.39	2.55	9.96	5.49
G-13	23767	0.45	62%	0.28	38%	33.38	0.73	9.34	2.35	9.93	4.41
G-14	31368	0.44	55%	0.36	45%	26.39	0.80	14.24	2.58	9.93	5.63
G-15	31248	0.45	55%	0.37	45%	31.91	0.82	12.71	2.64	9.91	5.61
G-16	31134	0.45	57%	0.34	43%	28.83	0.79	11.82	2.54	9.91	5.65
G-17	31145	0.45	54%	0.39	46%	32.14	0.84	12.51	2.56	9.94	5.57
G-18	29462	0.44	55%	0.36	45%	46.30	0.80	24.92	2.55	9.96	5.40
G-19	31290	0.44	53%	0.39	47%	38.52	0.83	8.28	2.54	9.97	5.63
G-20	30978	0.44	54%	0.37	46%	47.80	0.81	14.22	2.70	9.96	5.52
Average	30723	0.45	54.95%	0.37	45.05%	38.10	0.82	14.44	2.60	9.94	5.58

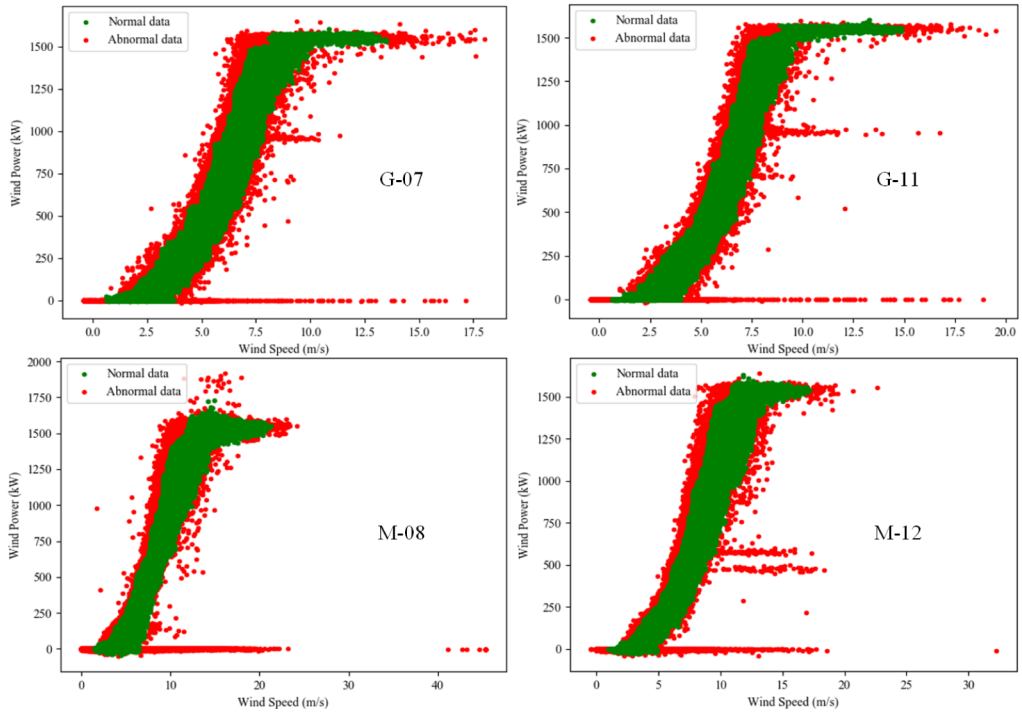

Fig. 7. Detection results of the proposed method on wind turbine G-07, G-11, M-08, and M-12.

algorithms on the GTX 1080Ti GPU for two wind farms are recorded and listed in Tables VII and VIII.

It is observed from Tables VII and VIII that the GPU-based LOF algorithm takes the longest time among all the three

TABLE IV
COMPUTATIONAL RESULTS OF DIFFERENT METHODS ON MATANG WIND FARM

WT	Amount of data	The proposed algorithm						MMO		LOF	
		Data Cleaning		Data Classification		Total					
		T_{CPU} (s)	Time Percentage	T_{CPU} (s)	Time Percentage	R (%)	T_{CPU} (s)	R (%)	T_{CPU} (s)	R (%)	T_{CPU} (s)
M-01	93230	0.42	27%	1.14	73%	22.20	1.56	12.04	4.06	9.99	16.67
M-02	90596	0.44	28%	1.11	72%	32.47	1.55	19.47	4.00	9.92	16.41
M-03	93383	0.45	28%	1.16	72%	22.85	1.61	11.40	4.07	9.93	16.24
M-04	93721	0.44	28%	1.16	72%	29.37	1.60	19.53	4.21	9.95	16.63
M-05	93394	0.43	28%	1.11	72%	28.75	1.54	20.00	4.18	9.94	17.12
M-06	92723	0.45	28%	1.16	72%	33.71	1.61	22.33	4.30	9.93	16.00
M-07	93702	0.51	29%	1.25	71%	65.27	1.76	19.13	4.21	9.93	16.77
M-08	92095	0.46	27%	1.24	73%	50.52	1.70	16.29	4.14	9.91	17.51
M-09	92937	0.46	28%	1.18	72%	41.83	1.64	18.85	4.16	9.89	16.43
M-10	92818	0.42	27%	1.12	73%	25.45	1.54	13.78	4.06	9.94	16.27
M-11	94218	0.44	28%	1.16	72%	18.78	1.60	8.93	4.12	9.94	18.88
M-12	94384	0.46	28%	1.16	72%	45.99	1.62	12.24	4.21	9.94	17.18
M-13	86437	0.46	32%	1.00	68%	31.89	1.46	16.49	4.04	9.94	15.08
M-14	88006	0.46	30%	1.08	70%	37.72	1.54	18.17	4.11	9.94	15.47
M-15	87084	0.45	31%	1.02	69%	47.49	1.47	14.25	4.06	9.95	15.47
M-16	87907	0.45	29%	1.08	71%	47.12	1.53	13.26	4.01	9.89	15.77
M-17	87198	0.43	29%	1.06	71%	28.62	1.49	16.78	3.99	9.95	15.99
Average	91402	0.45	28.53%	1.13	71.47%	35.88	1.58	16.06	4.11	9.93	16.46

TABLE V
SPECIFICATION OF THREE GPUS

Parameter	GTX 1050Ti	GTX 1080Ti	RTX 2080Ti
CUDA Version	10.1	10.1	10.1
CUDA Capability	6.1	6.1	7.5
CUDA Cores	768	3584	4352
Global Memory	4 GB	11 GB	11 GB
Constant Memory	64 KB	64 KB	64 KB
Shared Memory	48 KB	48 KB	48 KB
GPU Max Clock rate	1.62 GHz	1.72 GHz	1.54 GHz
Memory Clock rate	3504 MHz	5505 MHz	7000 MHz
Memory Bus Width	128-bit	352-bit	352-bit
Warp Size	32	32	32
Operating System	Ubuntu	Ubuntu	Ubuntu

TABLE VI
AVERAGE COMPUTING TIME AND SPEEDUP RATIOS OF THE PROPOSED METHOD ON THREE GPUS

GPU	Gaojiagou		Matang	
	T_{GPU} (s)	λ	T_{GPU} (s)	λ
1050Ti	0.0099	45.15	0.0099	45.33
1080Ti	0.0076	58.82	0.0076	59.08
2080Ti	0.0028	159.64	0.0028	160.36

TABLE VII
EXECUTION TIME OF DIFFERENT METHODS ON GTX 1080TI GPU FOR GAOJIAGOU WIND FARM

Wind Turbine	The proposed method	GPU-based MMO	GPU-based LOF
G-01	0.0075	0.017	0.20
G-02	0.0074	0.017	0.24
G-03	0.0078	0.017	0.19
G-04	0.0074	0.016	0.19
G-05	0.0076	0.016	0.19
G-06	0.0078	0.018	0.24
G-07	0.0075	0.016	0.19
G-08	0.0074	0.017	0.20
G-09	0.0075	0.017	0.20
G-10	0.0076	0.016	0.21
G-11	0.0077	0.018	0.20
G-12	0.0078	0.016	0.19
G-13	0.0078	0.017	0.18
G-14	0.0076	0.016	0.19
G-15	0.0076	0.016	0.19
G-16	0.0076	0.017	0.18
G-17	0.0075	0.017	0.20
G-18	0.0076	0.018	0.21
G-19	0.0078	0.016	0.18
G-20	0.0078	0.017	0.19
Average	0.0076	0.017	0.20

algorithms. The computing time of the GPU-based MMO algorithm is nearly 2.2 times as much as that of the proposed method. Computation results of both the CPU-based and GPU-based implementations confirm that the proposed method can achieve the best detection accuracy with the smallest computing time compared with the benchmarking algorithms.

Since large-scale wind energy optimization problems often contain large data sets, the scalability of the proposed method is examined based on different wind power data sets in the supplementary materials.

VI. WIND TURBINE POWER GENERATION PERFORMANCE EVALUATION

To accurately evaluate wind turbine power generation performance, WPC models are fitted using the processed data via

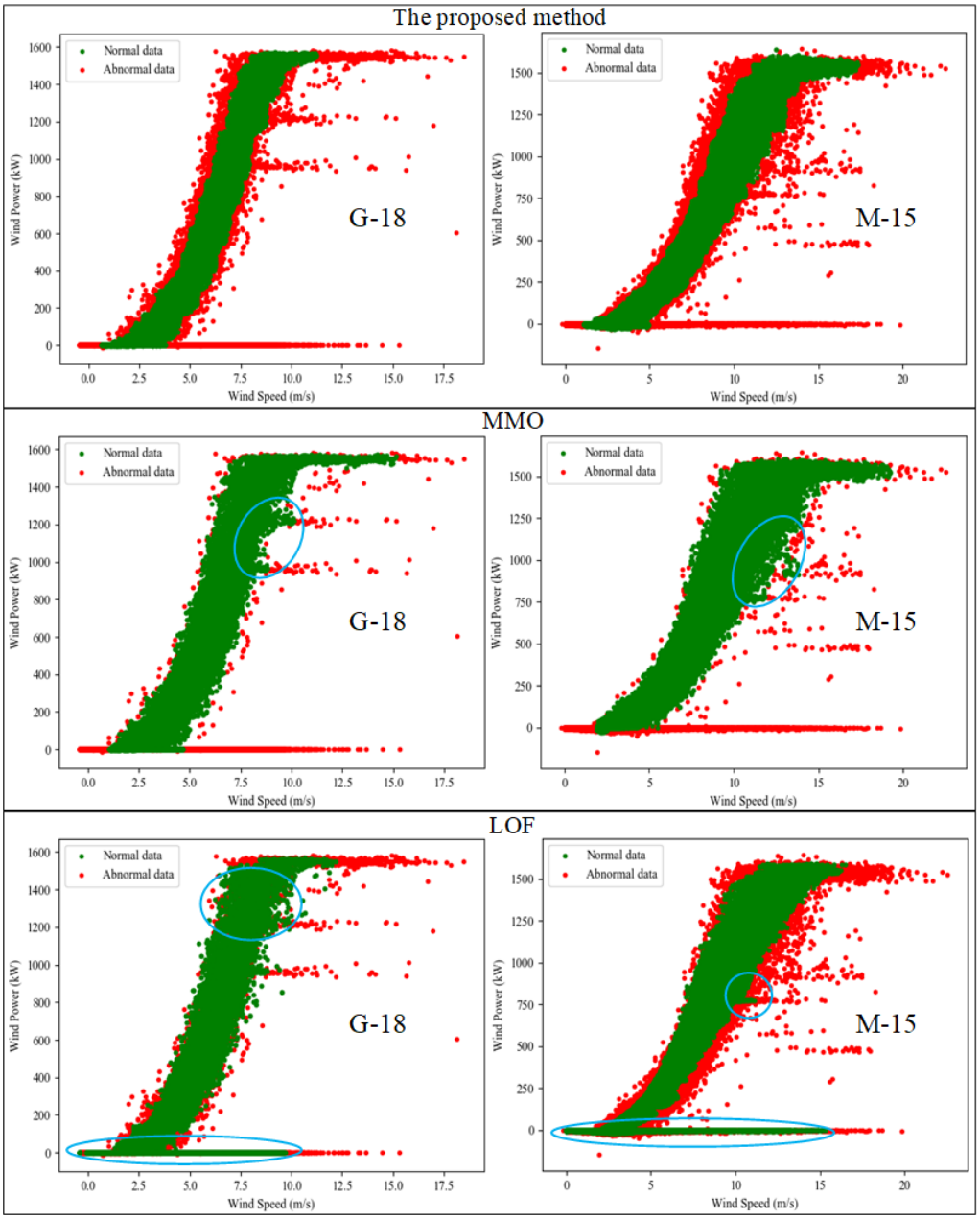

Fig. 8. Detection results of different methods on wind turbine G-18 and M-15.

MMO, LOF, and the proposed method. A logistic distribution model [40] is applied in this study as the WPC model. The logistic model is described as in the following equation:

$$P(v) = \alpha \frac{1 + me^{-\frac{v}{\sigma}}}{1 + ne^{-\frac{v}{\sigma}}} \tag{13}$$

where α, m, n, σ are four parameters that determine the curvature of WPCs. The model parameters are identified by using the Jaya algorithm [41] by minimizing the summed square of errors (SSE), shown in the following equation:

$$\text{SSE} = \sum_{i=1}^{N} [P_i - \hat{P}_i(v_i)]^2 \tag{14}$$

where P_i is the actual power and $\hat{P}_i(v_i)$ is the modeled power.

Based on the originally collected wind power data, the Gaussian process (GP) algorithm [42], and the Jaya algorithm [43] are also employed to develop WPC models. To compare the performance of the five models, two metrics, root mean square error (RMSE) and mean absolute error (MAE), are defined as the following equations:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} [P_{\text{ri}}(v_i) - P_{\text{ei}}(v_i)]^2} \tag{15}$$

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^{N} |P_{\text{ri}}(v_i) - P_{\text{ei}}(v_i)| \tag{16}$$

where P_{ri} is the power from the reference curve and P_{ei} is the power derived from the estimated curve.

TABLE VIII

EXECUTION TIME OF DIFFERENT METHODS ON GTX 1080TI GPU FOR MATANG WIND FARM

Wind Turbine	The proposed method	GPU-based MMO	GPU-based LOF
M-01	0.0075	0.018	0.76
M-02	0.0075	0.017	0.74
M-03	0.0076	0.016	0.75
M-04	0.0077	0.018	0.75
M-05	0.0077	0.016	0.73
M-06	0.0076	0.017	0.76
M-07	0.0076	0.017	0.74
M-08	0.0076	0.017	0.73
M-09	0.0075	0.018	0.74
M-10	0.0075	0.016	0.76
M-11	0.0074	0.018	0.74
M-12	0.0076	0.016	0.72
M-13	0.0077	0.017	0.75
M-14	0.0075	0.017	0.74
M-15	0.0077	0.016	0.75
M-16	0.0076	0.017	0.74
M-17	0.0076	0.016	0.73
Average	0.0076	0.017	0.74

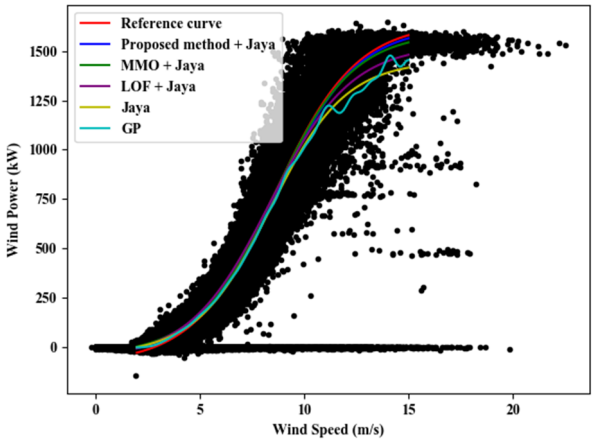

Fig. 9. WPCs of different models.

TABLE IX

RMSES AND MAES OF DIFFERENT MODELS

Model	RMSE	MAE
The proposed method + Jaya	14.45	13.63
MMO + Jaya	19.14	15.66
LOF + Jaya	48.21	38.19
Jaya	77.64	57.53
GP	79.12	56.72

Wind power data of wind turbine M-15 is used to fit each WPC model, depicted in Fig. 9, and 1000 wind speed values ranging from 3 to 15 m/s are employed to compute the RMSE and MAE values for each model. RMSEs and MAEs of different models are presented in Table IX.

From Table IX, it is observable that WPC models developed by using abnormal data cleaning algorithms obtain smaller RMSE and MAE values. Due to the influence of abnormal wind power data, the GP-based and Jaya algorithm-based methods can lead to WPCs with large deviations to the reference curves. Meanwhile, the proposed method outperforms MMO and LOF in cleaning abnormal wind power data and thus the most accurate WPC model is obtained via the proposed method. Therefore, the proposed method benefits WPC modeling, and wind turbine generation performance can be better measured and evaluated.

VII. CONCLUSION

This paper proposed a GPU-accelerated WPC abnormal data cleaning algorithm based on image processing for wind turbine power generation performance evaluation. The proposed algorithm included two stages, data cleaning and data classification. To further improve the execution speed, the data cleaning operation was executed in parallel on GPU via CUDA. The normal data points detected by the proposed method were applied to power generation performance evaluation. The main findings of this study are summarized as follows:

1) The proposed algorithm outperforms the MMO and LOF algorithms in terms of the highest detection rate and the shortest computing time.
2) The proposed algorithm can be further accelerated by using CUDA on GPUs and at least a 45× speedup ratio is obtained based on an entry-level GPU (Nvidia GTX 1050Ti).
3) Wind power data filtered by the proposed algorithm can be utilized to accurately measure the power generation performance.

In this study, only wind speed and power outputs are employed to evaluate wind turbine power generation performance. In future work, more operational parameters of wind turbines will be considered to develop a generation performance indication model. Meanwhile, the method will be further developed to eliminate the influences of outlier holes in WPC images.

REFERENCES

[1] I. Staffell and R. Green, "How does wind farm performance decline with age," *Renew. Energy*, vol. 66, pp. 775–786, Jun. 2014, doi: 10.1016/j.renene.2013.10.041.
[2] J. Nilsson and L. Bertling, "Maintenance management of wind power systems using condition monitoring systems—Life cycle cost analysis for two case studies," *IEEE Trans. Energy Convers.*, vol. 22, no. 1, pp. 223–229, Mar. 2007, doi: 10.1109/TEC.2006.889623.
[3] H. Long, L. Wang, Z. Zhang, Z. Song, and J. Xu, "Data-driven wind turbine power generation performance monitoring," *IEEE Trans. Ind. Electron.*, vol. 62, no. 10, pp. 6627–6635, Oct. 2015, doi: 10.1109/TIE.2015.2447508.
[4] Q. He, J. Zhao, G. Jiang, and P. Xie, "An unsupervised multiview sparse filtering approach for current-based wind turbine gearbox fault diagnosis," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 8, pp. 5569–5578, Aug. 2020, doi: 10.1109/TIM.2020.2964064.
[5] L. Wang, Z. Zhang, H. Long, J. Xu, and R. Liu, "Wind turbine gearbox failure identification with deep neural networks," *IEEE Trans. Ind. Informat.*, vol. 13, no. 3, pp. 1360–1368, Jun. 2017, doi: 10.1109/TII.2016.2607179.
[6] L. Wang, Z. Zhang, J. Xu, and R. Liu, "Wind turbine blade breakage monitoring with deep autoencoders," *IEEE Trans. Smart Grid*, vol. 9, no. 4, pp. 2824–2833, Jul. 2018, doi: 10.1109/TSG.2016.2621135.
[7] L. Wang, Z. Zhang, and J. Chen, "Short-term electricity price forecasting with stacked denoising autoencoders," *IEEE Trans. Power Syst.*, vol. 32, no. 4, pp. 2673–2681, Jul. 2017, doi: 10.1109/TPWRS.2016.2628873.
[8] L. Zheng, W. Hu, and Y. Min, "Raw wind data preprocessing: A data-mining approach," *IEEE Trans. Sustain. Energy*, vol. 6, no. 1, pp. 11–19, Jan. 2015, doi: 10.1109/TSTE.2014.2355837.
[9] S. Wang, Y. Huang, L. Li, and C. Liu, "Wind turbines abnormality detection through analysis of wind farm power curves," *Measurement*, vol. 93, pp. 178–188, Nov. 2016, doi: 10.1016/j.measurement.2016.07.006.

[10] M. Yesilbudak, "Partitional clustering-based outlier detection for power curve optimization of wind turbines," in *Proc. IEEE Int. Conf. Renew. Energy Res. Appl. (ICRERA)*, Nov. 2016, pp. 1080–1084, doi: 10.1109/ICRERA.2016.7884500.

[11] Y. Zhao, L. Ye, W. Wang, H. Sun, Y. Ju, and Y. Tang, "Data-driven correction approach to refine power curve of wind farm under wind curtailment," *IEEE Trans. Sustain. Energy*, vol. 9, no. 1, pp. 95–105, Jan. 2018, doi: 10.1109/TSTE.2017.2717021.

[12] X. Shen, X. Fu, and C. Zhou, "A combined algorithm for cleaning abnormal data of wind turbine power curve based on change point grouping algorithm and quartile algorithm," *IEEE Trans. Sustain. Energy*, vol. 10, no. 1, pp. 46–54, Jan. 2019, doi: 10.1109/TSTE.2018.2822682.

[13] A. Kusiak and A. Verma, "Monitoring wind farms with performance curves," *IEEE Trans. Sustain. Energy*, vol. 4, no. 1, pp. 192–199, Jan. 2013, doi: 10.1109/TSTE.2012.2212470.

[14] Y. Wang, D. G. Infield, B. Stephen, and S. J. Galloway, "Copula-based model for wind turbine power curve outlier rejection," *Wind Energy*, vol. 17, no. 11, pp. 1677–1688, Nov. 2014, doi: 10.1002/we.1661.

[15] X. Ye, Z. Lu, Y. Qiao, Y. Min, and M. O'Malley, "Identification and correction of outliers in wind farm time series power data," *IEEE Trans. Power Syst.*, vol. 31, no. 6, pp. 4197–4205, Nov. 2016, doi: 10.1109/TPWRS.2015.2512843.

[16] Y. Wang, Q. Hu, and S. Pei, "Wind power curve modeling with asymmetric error distribution," *IEEE Trans. Sustain. Energy*, vol. 11, no. 3, pp. 1199–1209, Jul. 2020, doi: 10.1109/TSTE.2019.2920386.

[17] L. Wang and Z. Zhang, "Automatic detection of wind turbine blade surface cracks based on UAV-taken images," *IEEE Trans. Ind. Electron.*, vol. 64, no. 9, pp. 7293–7303, Sep. 2017, doi: 10.1109/TIE.2017.2682037.

[18] L. Wang, Z. Zhang, and X. Luo, "A two-stage data-driven approach for image-based wind turbine blade crack inspections," *IEEE/ASME Trans. Mechtron.*, vol. 24, no. 3, pp. 1271–1281, Jun. 2019, doi: 10.1109/TMECH.2019.2908233.

[19] J. Wang, J. Wang, J. Shao, and J. Li, "Image recognition of icing thickness on power transmission lines based on a least squares Hough transform," *Energies*, vol. 10, no. 4, p. 415, Mar. 2017, doi: 10.3390/en10040415.

[20] H. Long, L. Sang, Z. Wu, and W. Gu, "Image-based abnormal data detection and cleaning algorithm via wind power curve," *IEEE Trans. Sustain. Energy*, vol. 11, no. 2, pp. 938–946, Apr. 2020, doi: 10.1109/TSTE.2019.2914089.

[21] Y. Su, F. Chen, G. Liang, X. Wu, and Y. Gan, "Wind power curve data cleaning algorithm via image Thresholding," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2019, pp. 1198–1203, doi: 10.1109/ROBIO49542.2019.8961448.

[22] M. J. Black and A. Rangarajan, "On the unification of line processes, outlier rejection, and robust statistics with applications in early vision," *Int. J. Comput. Vis.*, vol. 19, no. 1, pp. 57–91, Jul. 1996, doi: 10.1007/BF00131148.

[23] C. Riddell, P. Brigger, R. E. Carson, and S. L. Bacharach, "The watershed algorithm: A method to segment noisy PET transmission images," *IEEE Trans. Nucl. Sci.*, vol. 46, no. 3, pp. 713–719, Jun. 1999, doi: 10.1109/23.775604.

[24] H. Yu, Z. Li, and Z. Bao, "Residues cluster-based segmentation and outlier-detection method for large-scale phase unwrapping," *IEEE Trans. Image Process.*, vol. 20, no. 10, pp. 2865–2875, Oct. 2011, doi: 10.1109/TIP.2011.2138148.

[25] A. W. C. Liew, S. H. Leung, and W. H. Lau, "Fuzzy image clustering incorporating spatial continuity," *IEE Proc.–Vis., Image Signal Process.*, vol. 147, no. 2, pp. 185–192, Apr. 2000.

[26] S.-D. Kwon, "Uncertainty analysis of wind energy potential assessment," *Appl. Energy*, vol. 87, no. 3, pp. 856–865, Mar. 2010, doi: 10.1016/j.apenergy.2009.08.038.

[27] M. A. Mohamed, T. Jin, and W. Su, "An effective stochastic framework for smart coordinated operation of wind park and energy storage unit," *Appl. Energy*, vol. 272, Aug. 2020, Art. no. 115228, doi: 10.1016/j.apenergy.2020.115228.

[28] V. R. Parihar, "Image segmentation based on graph theory and threshold," *ICSES Trans. Image Process. Pattern Recognit.*, vol. 4, no. 4, pp. 61–82, Dec. 2018.

[29] R. Charifi, N. Essbai, A. Mansouri, and Y. Zennayi, "Comparative study of color image segmentation by the seeded region growing algorithm," in *Proc. IEEE 5th Int. Congr. Inf. Sci. Technol. (CiSt)*, Oct. 2018, pp. 279–284, doi: 10.1109/CIST.2018.8596399.

[30] T. O. Ting, J. Ma, K. S. Kim, and K. Huang, "Multicores and GPU utilization in parallel swarm algorithm for parameter estimation of photovoltaic cell model," *Appl. Soft Comput.*, vol. 40, pp. 58–63, Mar. 2016, doi: 10.1016/j.asoc.2015.10.054.

[31] L. Wang, Z. Zhang, C. Huang, and K. L. Tsui, "A GPU-accelerated parallel jaya algorithm for efficiently estimating li-ion battery model parameters," *Appl. Soft Comput.*, vol. 65, pp. 12–20, Apr. 2018, doi: 10.1016/j.asoc.2017.12.041.

[32] C. Huang, L. Wang, and L. L. Lai, "Data-driven short-term solar irradiance forecasting based on information of neighboring sites," *IEEE Trans. Ind. Electron.*, vol. 66, no. 12, pp. 9918–9927, Dec. 2019, doi: 10.1109/TIE.2018.2856199.

[33] T. Radil, P. M. Ramos, F. M. Janeiro, and A. C. Serra, "PQ monitoring system for real-time detection and classification of disturbances in a single-phase power system," *IEEE Trans. Instrum. Meas.*, vol. 57, no. 8, pp. 1725–1733, Aug. 2008, doi: 10.1109/TIM.2008.925345.

[34] Z. Chen, K. Xu, J. Wei, and G. Dong, "Voltage fault detection for lithium-ion battery pack using local outlier factor," *Measurement*, vol. 146, pp. 544–556, Nov. 2019, doi: 10.1016/j.measurement.2019.06.052.

[35] C.-X. Deng, G.-B. Wang, and X.-R. Yang, "Image edge detection algorithm based on improved canny operator," in *Proc. Int. Conf. Wavelet Anal. Pattern Recognit.*, Jul. 2013, pp. 168–172, doi: 10.1109/ICWAPR.2013.6599311.

[36] A. Q. Zhang, T. Y. Ji, M. S. Li, Q. H. Wu, and L. L. Zhang, "An identification method based on mathematical morphology for sympathetic inrush," *IEEE Trans. Power Del.*, vol. 33, no. 1, pp. 12–21, Feb. 2018, doi: 10.1109/TPWRD.2016.2590479.

[37] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying density-based local outliers," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, Dallas, TX, USA, 2000, pp. 93–104, doi: 10.1145/342009.335388.

[38] M. J. Thurley and V. Danell, "Fast morphological image processing open-source extensions for GPU processing with CUDA," *IEEE J. Sel. Topics Signal Process.*, vol. 6, no. 7, pp. 849–855, Nov. 2012, doi: 10.1109/JSTSP.2012.2204857.

[39] M. Alshawabkeh, B. Jang, and D. Kaeli, "Accelerating the local outlier factor algorithm on a GPU for intrusion detection systems," in *Proc. 3rd Workshop Gen.-Purpose Comput. Graph. Process. Units*, New York, NY, USA, Mar. 2010, pp. 104–110, doi: 10.1145/1735688.1735707.

[40] M. Lydia, A. I. Selvakumar, S. S. Kumar, and G. E. P. Kumar, "Advanced algorithms for wind turbine power curve modeling," *IEEE Trans. Sustain. Energy*, vol. 4, no. 3, pp. 827–835, Jul. 2013, doi: 10.1109/TSTE.2013.2247641.

[41] L. Wang, C. Huang, and L. Huang, "Parameter estimation of the soil water retention curve model with jaya algorithm," *Comput. Electron. Agricult.*, vol. 151, pp. 349–353, Aug. 2018, doi: 10.1016/j.compag.2018.06.024.

[42] R. K. Pandit and D. Infield, "Performance assessment of a wind turbine using SCADA based Gaussian Process model," *Int. J. Prognostics Health Manage.*, vol. 9, no. 1, p. 023, Jun. 2018.

[43] R. Jin, L. Wang, C. Huang, and S. Jiang, "Wind turbine generation performance monitoring with jaya algorithm," *Int. J. Energy Res.*, vol. 43, no. 4, pp. 1604–1611, Mar. 2019, doi: 10.1002/er.4382.

Zhongju Wang received the B.S. degree in computer science and technology from Zhengzhou University, Zhengzhou, China, in 2019. He is currently pursuing the M.Eng. degree in computer technology with the University of Science and Technology Beijing, Beijing, China.

Long Wang (Member, IEEE) received the M.Sc. degree (Hons.) in computer science from University College London, London, U.K., in 2014, and the Ph.D. degree in systems engineering and engineering management from the City University of Hong Kong, Hong Kong, in 2017.

He is currently an Associate Professor with the Department of Computer Science and Technology, University of Science and Technology Beijing, Beijing, China. His research interests include machine learning, computational intelligence, computer vision, and their industrial applications.

Dr. Wang is a member of the China Computer Federation (CCF), the CCF Technical Committee on Computer Vision, and the Program committee of World Automation Congress 2021. He was a recipient of the Hong Kong Ph.D. Fellowship, in 2014. He serves as an Associate Editor of IEEE ACCESS and an Academic Editor of *PLoS One*, as well as a Youth Editorial Committee Member of *Journal of Central South University*. He is also a Lead Guest Editor of data science-related special issues on *Frontiers in Neurorobotics*, *Intelligent Automation & Soft Computing*, and *Water*.

Chao Huang (Member, IEEE) received the B.Eng. degree in electrical engineering and automation from the Harbin Institute of Technology, Harbin, China, in 2011, M.S. degree in intelligent transport system from the University of Technology of Compiegne, Compiegne, France, in 2013, and the Ph.D. degree in systems engineering and engineering management from the City University of Hong Kong, Hong Kong, in 2017.

He is currently an Associate Professor with the Department of Computer Science and Technology, University of Science and Technology Beijing, Beijing, China. His research interests include machine learning and computational intelligence.