*Сучасний стан наукових досліджень та технологій в промисловості. 2026. № 2 (36)* *ISSN 2522-9818 (print)*
*Innovative technologies and scientific solutions for industries. 2026. No. 2 (36)* *ISSN 2524-2296 (online)*

UDC 004.415.2:004.89:621.548 DOI: https://doi.org/10.30837/2522-9818.2026.2.029

Lesia Dubchak, Nazar Vivchar, Oleg Savenko, Bohdan Derysh, Oleg Zastavnyy

# SOFTWARE TOOL FOR IMPLEMENTING AN INTELLIGENT METHOD FOR ANALYZING WIND TURBINE BLADE DEFECTS WITH LIMITED DATASET

The **subject** of the study is methods and software tools for intelligent diagnostics of wind turbine blade defects based on neuro-fuzzy models capable of operating effectively in conditions of limited, fuzzy, or partially defined input data. The **goal** of the study is to develop a software implementation and conduct an experimental study of a neuro-fuzzy Wang–Mendel network for classifying wind turbine blade defects, and to justify its feasibility as the basis of an intelligent system for monitoring the technical condition of energy facilities. The main **tasks** are to analyze modern methods of defect classification, develop the architecture of a neuro-fuzzy model, create a software package for training and classification, implement an algorithm for automatically forming a fuzzy rule base, optimize the parameters of membership functions, and conduct experimental studies to assess the effectiveness of the proposed approach. The study used **methods** of fuzzy logic, neuro-fuzzy systems, mathematical modeling, data classification, parameter optimization, and software implementation of intelligent systems in Python using modern data processing and numerical computing libraries. The Wang–Mendel algorithm provides automatic formation of fuzzy rules based on training data and adaptation of model parameters to increase classification accuracy. As a **result**, an intelligent software complex for classifying wind turbine blade defects was developed, which provides effective data processing and decision-making based on fuzzy rules. Experimental studies were conducted on a training sample of 160 records and a test sample of 64 records, describing the probabilities of the presence of various types of defects. **Conclusions:** according to the training results, an accuracy of 91% was achieved on the training sample and 94% on the test sample with a low level of mean square error, which confirms the high efficiency, generalization ability, and practical suitability of the proposed model for use in intelligent systems for monitoring the technical condition of wind turbines in conditions of limited data and resources.

**Keywords:** wind turbines; defects; neuro-fuzzy systems; Wang–Mendel; Python.

## Introduction

Over the past 5–10 years, wind power has shown steady and rapid growth worldwide and in Europe. Global wind power capacity is expected to increase by 50% in the coming years [1, 2]. Europe is a leader in the development of this sector, with an average annual growth rate of 15–16% over the past two decades [3, 4]. The largest capacities are concentrated in Germany, Spain, the United Kingdom and France, which together provide more than 60% of all wind installations on the continent [5].

Wind power is expected to remain a key driver of the global energy transition between 2030 and 2050, driven by technological innovation, cost reductions and supportive policies [6, 7]. Mid-term projections also predict a significant reduction in the cost of wind power: by 2050, the cost could fall by 37–49%, reaching \$20–30/MWh for onshore and \$40–60/MWh for offshore wind farms [8, 9].

At the same time, the rapid growth of the industry places new demands on the reliability and efficiency of wind turbine operation. The economic performance of energy companies, the competitiveness of renewable energy, and the success of the transition to a low-carbon economy directly depend on the performance of the equipment.

Wind turbine failures have a direct and significant impact on both operating costs and performance. Failures increase maintenance requirements, reduce energy production, and shorten the life of the equipment, which together lead to increased costs, reduced efficiency, and consequently increased electricity costs [10, 11]. Minor failures, such as blade surface erosion, are a major cause of unplanned repairs, the cost of which is much higher than the cost of repairs for major structural failures [12].

Thus, the development of wind energy requires increased attention to the condition of turbines and their maintenance methods. Ensuring the stability of equipment operation is becoming an important prerequisite for further growth of the industry, which emphasizes the relevance of research in the field of diagnostics of wind turbine blade defects.

The research aims to develop, implement, and investigate the effectiveness of a Wang–Mendel neuro-fuzzy network for the task of classifying defects in wind turbine blades.

ISSN 2522-9818 (print) Сучасний стан наукових досліджень та технологій в промисловості. 2026. № 2 (36)
ISSN 2524-2296 (online) Innovative technologies and scientific solutions for industries. 2026. No. 2 (36)

To achieve the stated objective, the following research tasks must be addressed:

1. To analyze modern methods and approaches for diagnostics and classification of wind turbine blade defects, including machine learning, deep learning, and neuro-fuzzy systems, and to identify their advantages and limitations.
2. To develop the model of an intelligent defect classification system based on the Wang–Mendel neuro-fuzzy network, capable of automatically generating a fuzzy rule base from training data.
3. To implement a software system for wind turbine blade defect classification using the Python programming language, incorporating data preprocessing, fuzzification, fuzzy inference, and adaptive parameter optimization.
4. To train the neuro-fuzzy model using experimental data and determine the optimal parameters of membership functions and the structure of the fuzzy rule base.
5. To conduct experimental studies to evaluate the performance of the developed model using training and testing datasets, and to assess its classification accuracy, error rate, and generalization capability.
6. To perform a comparative analysis of the proposed approach with existing defect classification methods and to justify the feasibility of applying the Wang–Mendel neuro-fuzzy network in intelligent monitoring systems for wind turbine condition assessment.

## Analysis of last achievements and publications

The development of wind energy is accompanied by increasing requirements for the reliability of wind turbines, in particular for their key structural elements – blades. Traditional diagnostic methods based on visual inspection or classical signal analysis algorithms are often laborious, expensive and limited in accuracy.

In this regard, machine learning is increasingly being used to improve the efficiency of blade defect detection. Current research applies various machine and deep learning methods using visual and sensory data for more accurate identification and classification of damage. Convolutional neural networks, in particular Xception, ResNet-50, AlexNet, VGG-19, as well as specialized architectures, have demonstrated high efficiency in defect classification tasks: the use of the Xception model with transfer learning provided accuracy up to 99.92% on small samples and up to 100% on large ones [13].

The development of real-time object detection architectures has led to the widespread use of advanced YOLO models, such as YOLOv5s, YOLOX, and YOLOv8n. Their combination with attention mechanisms, feature fusion technologies, and efficient loss functions has significantly increased the average accuracy and speed of detection [14]. Signal analysis methods are actively used to detect internal defects, including cracks and delamination: guided waves, wavelet transforms, feature extraction algorithms in combination with k-nearest neighbors' classifiers, support vector methods, and random forest. Such approaches provide classification accuracy of up to 97% [15]. Additionally, adaptive neuro-fuzzy inference systems (ANFIS) and Gaussian processes are used, which reduce computational costs and allow for early damage detection with an accuracy of up to 91% [16].

Key advantages of these methods include high accuracy and speed of data processing, the ability to detect a wide range of damage, and the possibility of integration with unmanned aerial vehicles for remote monitoring [15, 16]. The use of weakly supervised learning and drones for image collection can significantly reduce the need for manual annotation and reduce the risk of human error. Integration with signal processing systems and the use of optimized models make it possible to perform large-scale automated maintenance with minimal resource costs [17].

At the same time, these approaches have certain limitations. Deep neural networks require large volumes of well-annotated data, which complicates their application in practice. Classical algorithms remain vulnerable to the detection of small or hidden defects, although modern improvements partially solve this problem. The computational requirements are also significant, but the use of lightweight models and effective feature selection methods allows for reducing the load [18]. A current research direction is to increase the resistance of models to a decrease in the quality of input data (low resolution, overlapping objects) and to develop more effective methods of merging and supplementing data.

Despite the high performance of modern machine learning methods mentioned earlier, they have a number of limitations, including significant data requirements, the need for careful annotation, and high computational load. In practical wind energy applications, only limited or partial data of small volume are often available, which reduces the effectiveness of classical neural networks

*Сучасний стан наукових досліджень та технологій в промисловості. 2026. № 2 (36)* *ISSN 2522-9818 (print)*
*Innovative technologies and scientific solutions for industries. 2026. No. 2 (36)* *ISSN 2524-2296 (online)*

or deep learning methods. In this context, approaches that can work with incomplete or fuzzy input data without losing the accuracy of defect classification are of particular importance.

Given the successful application of fuzzy logic in medicine, robotics and other areas [19–21], its use in wind turbine damage diagnostics is relevant. Neural-fuzzy networks, which combine the capabilities of fuzzy inference systems and artificial neural networks, demonstrate high efficiency in cases where the data are limited or contain uncertainty. Of particular interest is the Wang-Mendel neuro-fuzzy network, which provides flexible formation of inference rules based on input data and allows achieving high accuracy even on small samples [19]. The functioning of such systems is based on fuzzification – the transformation of clear input values into fuzzy sets using membership functions and the reverse process of defuzzification, which allows obtaining one clear value at the output. This provides an interpreted representation of knowledge in the form of a system of fuzzy rules, bringing the decision-making process closer to the logic of human experts. Adding a neural component allows for the implementation of a learning mechanism through which the parameters of the membership functions can adapt and the fuzzy inference rules can change to increase the accuracy of solving the problem.

## Modification of the Wang–Mendel neuro-fuzzy network

Fuzzy inference systems and neural networks have similar principles of operation. As inference systems, they require training, the formulation of a certain representation of the knowledge obtained during training, and an algorithm for processing new data using certain rules. The similarity is as follows:

1. Neural networks are trained on input data, while fuzzy systems mostly rely on an expert who sets the inference rules.
2. The "knowledge" of a neural network is stored in the format of weight coefficients, and of a fuzzy system, in the form of rules represented by membership functions.
3. Both systems process input data based on acquired knowledge and a predefined algorithm.

Fuzzy systems and neural networks operate as dynamic systems capable of parallel and distributed data processing. Both approaches are excellent at approximating functions without relying on predefined mathematical models, instead drawing knowledge directly from sample data. These systems are not only similar in their adaptive nature but also provide unique advantages when applied to real-world scenarios [6].

A hybrid model that combines the concepts of fuzzy logic and artificial neural networks is called a neuro-fuzzy network. Due to the peculiarity of this combination, there is a system that is based on the strengths of both methods and allows you to work effectively with inaccurate, partial or limited information about complex systems. A great advantage of such an architecture is the combination of the interpretability of a fuzzy inference system with the ability to use training methods familiar to "classical" neural networks. Due to this, the accuracy of the hybrid model is better than that of a simple fuzzy inference system, and such a system is also better able to handle classification tasks based on large volumes of data.

One type of neural fuzzy network is the Wang–Mendel neural fuzzy network, proposed by Jerry Mendel and Li-Xin Wang in 1992 [8]. The architecture of such a neural network consists of four layers:

- fuzzification layer;
- aggregation layer;
- linear layer;
- defuzzification layer.

The fuzzification layer, which also acts as an input layer, takes as input a data vector $x_1, x_2, ..., x_N$ of length $N$ and fuzzified each variable separately to formulate a fuzzy set $A$ calculating the value $L$ of each of the membership functions $\mu_l\left(x_n\right)$ $\left(l = 1, 2, ...L;\ n = 1, 2, ...N\right)$.

The generalized Gaussian function is used as the membership function:

$$\mu_A\left(x_n\right) = \frac{1}{1 + \left(\frac{x_n - c_n}{\sigma_n}\right)^2} . \qquad (1)$$

To initialize the fuzzification layer, it is necessary to determine the number $L$ of membership functions, and the parameters $c_n$ and $\sigma_n$ $\left(n = 1, 2, \ldots, N\right)$, which will be refined during the training process to obtain a better refinement of the final prediction of the network. In some cases, the initial values of the parameters $c_n$ and $\sigma_n$ and the number of membership functions can be either calculated automatically or statically specified before the network training begins. We will consider the case with predefined parameters of the first layer. At the output of the fuzzification layer, $L$ vectors of

ISSN 2522-9818 (print) Сучасний стан наукових досліджень та технологій в промисловості. 2026. № 2 (36)
ISSN 2524-2296 (online) Innovative technologies and scientific solutions for industries. 2026. No. 2 (36)

dimension $N$ are obtained, where each element is calculated by the formula

$$\mu_{lr}\left(x_r\right)=\frac{1}{1+\left(\frac{x_r-c_{lr}}{\sigma_{lr}}\right)^2} \quad \left(l=1,2,...L;\ r\ =\ 1,2,...N\right), \quad (2)$$

where $x_r$ – the $r$-th element of the input feature vector, $c_{lr}$ – the center of the Gaussian membership function, $\sigma_{lr}$ – the width (variance) of the function.

The aggregation layer performs the composition of the membership function values $L$ calculated by the previous layer, formulating the inference rules. The calculation is carried out according to the formula:

$$\mu_l\left(x\right)=\prod_j^N \mu_{lj}\left(x_j\right)\ . \quad (3)$$

The aggregation layer does not contain parameters that could change during training.

The linear layer is parametric and contains $L$ synaptic weights $w_1, w_2, ..., w_l$ whose values will adapt during training. The initial value of these weights is chosen randomly from the interval $\left[-1;1\right]$. This layer is used to aggregate inference rules according to the formulas $f_1=\sum_l^L w_l\cdot\mu_l\left(x\right),\ \ f_2=\sum_l^L \mu_l\left(x\right)$.

The fourth layer is responsible for data defuzzification and obtaining the system output signal according to the formula

$$y\left(x\right)=f_1/f_2\ . \quad (4)$$

The value $y\left(x\right)$ obtained at the end will be the prediction of the Wang–Mendel neural fuzzy network. Thus, the task of the network is that for any pair of data $\left\langle x,d\right\rangle$, the prediction of the neural network $y\left(x\right)$ corresponds to the expected value $d$ .

Several common algorithms are used to train a Wang–Mendel neural fuzzy network, the most common of which is based on the principle of minimizing the objective function. This method is similar to the method of training conventional neural networks and consists in selecting such values of weights $w_1, w_2, ..., w_L$ and parameters $c_n$ and $\sigma_n$ to obtain the most accurate result. For $K$ training samples, the objective function can be specified as

$$E=\frac{1}{2}\sum_1^K\left(y\left(x^k\right)-d^k\right), \quad (5)$$

where $x^k$ – input training vector, $y\left(x^k\right)$ – prediction of the fuzzy neural network for vector $x^k$, $d^k$ – expected value of the network output for vector $x^k$ .

Training a Wang–Mendel fuzzy neural network using the objective function minimization algorithm occurs through the alternation of two stages:

1. For the fixed parameters $c_n$ and $\sigma_n$ the first layer of the network, the values $w_1, w_2, ..., w_L$ of the synaptic weights of the third layer are calculated.
2. For the fixed values of the synaptic weights of the third layer, the values of the parameters $c_n$ and $\sigma_n$ the first layer are calculated.

At the first stage of training a neural network, training $K$ samples $\left\langle x^k, d^k\right\rangle$ are selected. Based on these samples, a vector $D=\left\{d^1, d^2, ..., d^k\right\}$, $\left(k=1,2,...,K\right)$ of expected network responses is formed. Based on the obtained values, a system of $K$ linear equations is formulated:

$$PV\cdot W=D\ , \quad (6)$$

where $W$ – the desired vector of values $w_1, w_2, ..., w_N$ of synaptic weights of the third layer, $PV$ – the matrix of coefficients, where each value of the element of each row corresponds to the coefficient calculated for the corresponding value of the input vector of the corresponding sample, which can be expressed as

$$PV_r^{\ k}=\prod_j^N \mu_{rj}\left(x_j^k\right)\Big/\sum_{l=1}^L\prod_j^N \mu_{rj}\left(x_j^k\right). \quad (7)$$

In this case, a much larger number of training samples $K$ is selected than the dimension of the input vector $N$, accordingly, the number of rows of the coefficient matrix $PV$ will be much larger than the number of its rows.

From equation (6) we can find the value of the vector $W$ and represent it as an equation

$$W=PV^+\cdot D\ , \quad (8)$$

where $PV^+$ – pseudo-inverse matrix to the $PV$ matrix. The pseudo-inverse matrix can be searched for by any of the corresponding algorithms, but this does not affect the training of the network. Therefore, by solving the system of linear equations, the vector $W$ of synaptic weights of the third layer will be obtained. After this, the stage of synaptic weight refinement is considered complete.

At the second stage, to calculate the values of the synaptic weights of the third layer of the neural network,

*Сучасний стан наукових досліджень та технологій в промисловості. 2026. № 2 (36)* *ISSN 2522-9818 (print)*
*Innovative technologies and scientific solutions for industries. 2026. No. 2 (36)* *ISSN 2524-2296 (online)*

the parameters $c_n$ and $\sigma_n$ (i.e., the coefficients of the Gaussian membership function) of the first layer are refined. For this, the gradient descent method is used, for which the values and on the training cycle can be expressed as

$$\begin{cases} c_{ij}\left(k+1\right) = c_{ij}\left(k\right) - \eta_c \cdot \dfrac{\partial E\left(k\right)}{\partial c_{ij}} \\ \sigma_{ij}\left(k+1\right) = \sigma_{ij}\left(k\right) - \eta_c \cdot \dfrac{\partial E\left(k\right)}{\partial \sigma_{ij}}, \end{cases} \tag{9}$$

where $\eta_c$ and $\eta_\sigma$ are coefficients that determine the speed of parameter adaptation $c_n$ and $\sigma_n$, respectively, $\partial E\left(k\right)/\partial c_{ij}$ and $\partial E\left(k\right)/\partial \sigma_{ij}$ are partial derivatives of network errors.

After refinement $c_n$ and $\sigma_n$, the learning process is restarted from the first stage and repeated until either the desired accuracy is achieved or all parameters (of the first and third layers are stabilized). When implementing this learning algorithm in practice, the first stage (refinement of the weights of the third layer by solving a system of linear equations) is of the greatest importance, which is performed in one step, and the second stage is secondary. The second stage of learning is repeated a significant number of times with each new iteration of learning.

## Software implementation

The algorithmic logic of the method implementation in the software system can be formalized in the form of generalized pseudocode, which reflects the complete system workflow from data acquisition to obtaining the classification result, as shown in Figure 1.

```
Input: Training dataset D_tr, Testing dataset D_te
Output: Predicted class labels for D_te

Read D_tr, D_te from CSV files
Preprocess data (numeric conversion, NaN removal, shuffling)
Normalize all input variables to [0,1]
Define fuzzy partitions for each input variable
Initialize empty rule base RB
for each sample (x, y) in D_tr do
    for each input variable x_i do
        determine fuzzy term A_i with max membership μ(A_i, x_i)
    end for
    compute rule weight w = Π μ(A_i, x_i)
    form rule R: IF x_1 is A_1 AND … AND x_n is A_n THEN class y
    if conflict with existing rule in RB then
        keep rule with larger weight
    else
        add R to RB
    end if
end for
for each sample x in D_te do
     compute activation of all rules in RB
     aggregate activations per class
     assign class with maximum aggregated a[illegible]ation
end for
```

**Fig. 1.** Pseudocode of the fuzzy rule processing method implementation

The program code implementing the modified Wang–Mendel neuro-fuzzy network is structured as separate functional blocks responsible for data input, normalization, dataset diagnostics, fuzzy model construction, and implementation of the Wang–Mendel algorithm. In particular, the code includes functions for loading and analyzing training and testing datasets, normalizing numerical features, generating fuzzy partitions, and auxiliary procedures for displaying statistical information about the data, which allows monitoring the correctness of preprocessing stages (Fig. 2).

From a data flow perspective, the software system forms a closed processing pipeline in which the results of each stage are directly used in the subsequent stage, and the output decisions are transferred to the evaluation subsystem for generating the confusion matrix and quantitative performance metrics. The software architecture has a modular structure and includes subsystems for data input and storage, preprocessing and normalization, fuzzy model definition, Wang–Mendel rule generation, fuzzy inference mechanism, and result evaluation and visualization. This separation of functional responsibilities ensures the extensibility of the software system and provides the foundation for its further integration with other intelligent methods or hardware-based data acquisition systems.

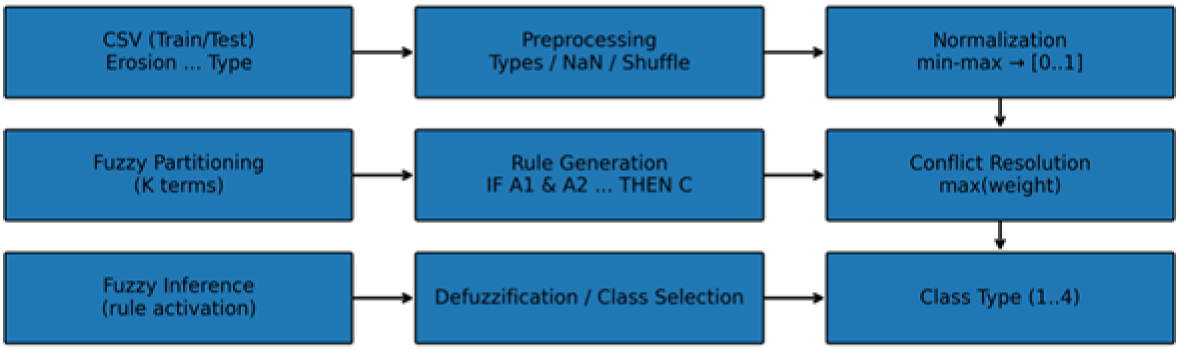


**Fig. 2.** Main modules of the Wang–Mendel network software implementation

Two data sets were used for training: training (160 records) and test (64 records). Each record was a vector of five elements: the first four corresponded to the probabilities of the presence of a certain defect (from 0 to 1), and the fifth element contained the actual class, one of four possible states of the blade. Three types of defects were taken into account: erosion, corrosion and crack, as well as a class corresponding to the absence of damage. Accordingly, each record clearly belonged to one of the four classes that determined the real state of the wind turbine. Before feeding into the model, the data were normalized. An example of the initial (before normalization) records for each of the classes is given in Table 1.

**Table 1.** *Example of input data sample for training the proposed neuro-fuzzy network*

| Probability of erosion | Corrosion probability | Probability of a crack | Probability of normal condition |
|---|---|---|---|
| 0.889 | 0.0134 | 0.0972 | 0.0005 |
| 0.0694 | 0.4155 | 0.2806 | 0.2345 |
| 0.1482 | 0.2308 | 0.4903 | 0.1308 |
| 0.0009 | 0.0568 | 0.055 | 0.8872 |
| 0.889 | 0.0134 | 0.0972 | 0.0005 |
| 0.0694 | 0.4155 | 0.2806 | 0.2345 |

The Wang–Mendel fuzzy neural network was trained using the objective function minimization algorithm by alternately optimizing the parameters of the membership functions of the first layer and the synaptic weights of the third layer.

The implementation of the Wang–Mendel algorithm was carried out using the interpreted object-oriented programming language Python in the PyCharm development environment. The entire architecture of the neural network was created without the use of third-party libraries. To perform individual mathematical operations, analyze data and display them, auxiliary libraries were used: math (simple mathematical operations), time (measurement of program execution time), pandas (data manipulation and analysis), numpy (performance of matrix operations and high-level mathematical operations), matplotlib (display of data in a graphical representation), tqdm (display of learning progress).

The optimal number of membership functions was determined empirically: increasing their number generally improved accuracy, but after a certain threshold, the efficiency began to decrease, while the training time continued to increase. The best results were obtained with 30 membership functions. The initial centers of the Gaussian functions were evenly distributed in the interval [1, 4], and the widths were fixed at 0.2. Since these parameters were optimized during the training process, their initial choice did not have a critical impact on the final result and was based on expert judgment.

The network was trained for 10 epochs with an early stop mechanism: if the classification accuracy and

*Сучасний стан наукових досліджень та технологій в промисловості. 2026. № 2 (36)* *ISSN 2522-9818 (print)*
*Innovative technologies and scientific solutions for industries. 2026. No. 2 (36)* *ISSN 2524-2296 (online)*

the standard deviation remained unchanged for three consecutive epochs, training was stopped. As a result, the model achieved an accuracy of 91% with a standard deviation of 0.0975. The training process was completed after four full epochs according to the early stop criterion. The total training time was 20 minutes 39 seconds.

## Experimental studies

The program available at the GitHub repository link [22] implements the Wang–Mendel method, which combines the principles of fuzzy logic and heuristic analysis algorithms to create a fuzzy control system, particularly for data classification and solving optimization problems. A detailed description of its main components is provided below.

The program is implemented in Python and its logic involves sequential data processing, starting with loading the necessary libraries and data sets, followed by their preliminary preparation, which includes cleaning from gaps, normalization and standardization of features to ensure correct operation of the algorithm and increase the stability of the learning process. The central element of the program is the implementation of the Wang–Mendel algorithm, which forms a fuzzy classification model based on training data, automatically generating a database of fuzzy rules that reflect the patterns between the input features and the corresponding classes. To increase the efficiency of the model, a parameter optimization mechanism is provided, which allows you to adapt the membership functions and other system parameters in order to achieve the optimal ratio between classification accuracy and computational costs. The main functions of the software tool provide a full cycle of model operation, including the training procedure based on the training sample and the formation of predictions for new data, which allows using the developed system as a universal classification tool.

The program is organized in an interactive Jupyter Notebook environment, which provides the ability to step-by-step execute the algorithm, monitor intermediate results, and analyze the model's effectiveness in real time. After executing the corresponding code blocks, the user receives numerical indicators of the classification quality, in particular, the accuracy and error values, which allows you to assess the effectiveness of the model and its generalization ability. In addition to numerical results, the program generates graphical representations of the classification results, including error matrices and scatter plots, which provides a visual representation of the correspondence between the true and predicted classes and allows you to assess the nature of the error distribution. In particular, Fig. 3 shows a scatter plot of the true and predicted classes for the test sample, which demonstrates a high level of correspondence between the classification results and the actual values, which confirms the effectiveness of the proposed approach.

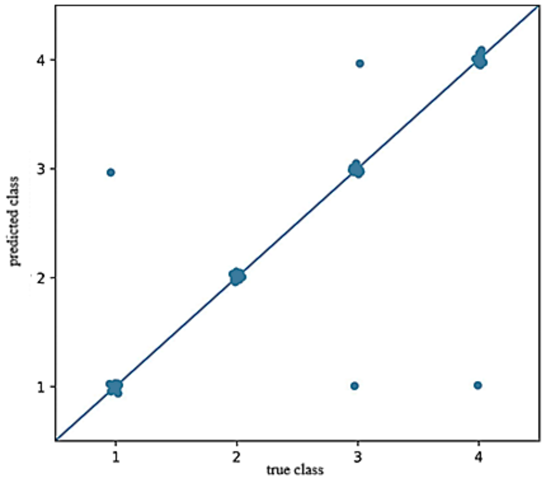


**Fig. 3.** Scatter plot of true and predicted classes for the test sample

The principle of operation of the software tool is based on the use of fuzzy logic and heuristic learning, which allows you to work effectively in conditions of uncertainty and incomplete data. The fuzzy model forms a system of rules of the "if–then" type, which allows you to perform classification based on the degree of belonging of the input data to the corresponding fuzzy sets, ensuring a smooth transition between classes and increased resistance to noise. The heuristic nature of the algorithm allows you to automatically form the optimal structure of the model based on the analysis of training data without the need for complex iterative training procedures, which reduces computational costs and increases the speed of the system. Thanks to this, the user has the opportunity to adapt the model parameters in accordance with the specifics of a particular task, which ensures the versatility of the software tool and the possibility of its use for different types of data.

Experimental studies were conducted on a test sample of 64 records, each record containing four features corresponding to the estimated probabilities of erosion, corrosion, crack, or no defect, and belonging to one of four classes. The results obtained demonstrate the high efficiency of the developed software tool: the classification accuracy on the test sample reached 94% accuracy at $RMSE = 0.0686$. This indicates a good generalization ability of the model and its efficiency when working with new data, and also confirms the feasibility of using the Wang–Mendel method as the basis of an intelligent defect classification system in conditions of limited amounts of training data and the need to ensure high accuracy and interpretability of the results.

## Comparison with analogues

A comparative analysis of modern approaches to defect classification shows that their effectiveness is largely determined by the balance between accuracy, training data volume requirements, computational resources, and interpretability of the results obtained. As shown in Fig. 4, the maximum accuracy values are demonstrated by deep convolutional neural networks, in particular Xception TL, which provide up to 99.92% of correct classifications. The main advantage of such models is the ability to automatically generate informative features without the need for manual design. At the same time, their practical application is significantly limited by high requirements for the training sample volume, significant computational costs, and complexity of implementation, which complicates their use in embedded or resource-limited monitoring systems.

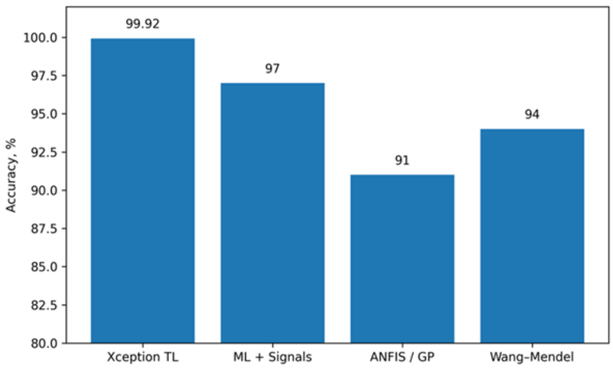


**Fig. 4**. Comparison of accuracy with analogues

Signal analysis approaches using classical machine learning methods demonstrate somewhat lower but consistently high accuracy rates, reaching 97%. Their characteristic feature is lower computational costs compared to deep neural networks and the possibility of effective use with limited resources. However, the effectiveness of such methods significantly depends on the quality of sensor data and the correctness of feature vector formation, which reduces their versatility and complicates their application in variable or noisy environments.

Methods based on neuro-fuzzy systems, in particular ANFIS and Gaussian processes, demonstrate increased efficiency when working with small samples and provide accuracy at the level of about 91%. Their key advantage is the ability to work under uncertainty and take into account the fuzzy nature of the input data. However, the complexity of parameter tuning, the increase in computational complexity with increasing dimensionality of the problem, and limited scalability may limit their use in real monitoring systems with large data streams.

The proposed approach based on the Wang–Mendel method demonstrates classification accuracy at the level of 94%, which exceeds the indicators of classical neuro-fuzzy systems and is comparable to more complex machine learning methods, while maintaining significantly lower requirements for resources and the amount of training data. The key feature of this method is the formation of an explicit base of fuzzy rules without the need for complex iterative training procedures, which ensures high computational efficiency and stability of results even with a limited training sample. Unlike deep neural networks, the proposed approach does not require large arrays of labeled data and can function effectively in conditions of limited information, which is critically important for monitoring tasks of technical objects, where obtaining large samples is a difficult or expensive process.

An additional advantage of the Wang–Mendel method is its interpretability, since the classification results are formed on the basis of a clearly defined base of fuzzy rules, which allows for the analysis of the decision-making logic and ensures the transparency of the system. This is a fundamental difference from deep neural networks, which function as a "black box" and do not allow for a direct explanation of the process of forming the result.

Thus, the proposed method provides an optimal compromise between classification accuracy, computational efficiency, ability to work with small samples, and interpretability of results. This confirms the feasibility of its use as the basis of an intelligent defect classification system in resource-limited and real-world operating conditions.

## Conclusions

This paper analyzes the current state and trends in wind energy development, demonstrating the need to improve the reliability and efficiency of wind turbines, in particular their blades. It is shown that defects in structural elements significantly affect the productivity, economic performance and duration of equipment operation, and therefore the competitiveness of the entire industry.

A review of existing diagnostic methods revealed that traditional approaches are inferior to modern machine and deep learning algorithms in terms of accuracy and speed. Still, the latter require large volumes of qualitatively annotated data and have high computational costs. In this regard, a promising direction is the use of neuro-fuzzy systems that can operate effectively under data incompleteness or uncertainty.

Within the framework of the study, a neuro-fuzzy network based on the Wang–Mendel method was implemented for the classification of defects in wind turbine blades. The developed model combines the interpretability of fuzzy inference systems with the adaptability of artificial neural networks, which made it possible to achieve high classification accuracy with a relatively small amount of training data. According to the results of the experiments, the training accuracy was 91%, and on the test set – 94% with a standard deviation of 0.0686, which confirms the effectiveness of the proposed architecture for defect diagnosis tasks.

The obtained results indicate the feasibility of using Wang–Mendel neuro-fuzzy networks in wind turbine technical condition monitoring systems, especially in cases of limited or partially fuzzy data. Further research should be directed towards expanding the sample size, optimizing membership functions, and combining neuro-fuzzy systems with computer vision and deep learning methods to increase the model's robustness to noise and input variability.

### Conflict of interest

The authors declare that they have no conflicts of interest, including financial, personal, authorship, or any other conflicts that could influence the research or the results published in this article.

### Funding

This research was supported by the Ministry of Education and Science of Ukraine and funded by the European Union's external aid instrument as part of Ukraine's commitments under the EU Framework Program for Research and Innovation "Horizon 2020". The work was carried out as part of the research project "Intelligent Defect Detection System for Green Energy Facilities Using UAVs", state registration number 0124U004665 (2023–2026).

### Data availability

Data will be provided upon reasonable request

### Use of artificial intelligence

The authors confirm that they did not use artificial intelligence technology in the creation of this paper.

### References

1. Arshad, M., O'Kelly, B. (2019), "Global status of wind power generation: theory, practice, and challenges", *International Journal of Green Energy*, Vol. 16, pp. 1073–1090. DOI: https://doi.org/10.1080/15435075.2019.1597369
2. Bukovsky, M., Barthelmie, R., Leung, L., Pryor, S. and Sakaguchi, K. (2020), "Climate change impacts on wind power generation". *Nature Reviews Earth & Environment*, Vol. 1, pp. 627–643. DOI: https://doi.org/10.1038/s43017-020-0101-7
3. Hatziargyriou, N., Zervos, A. (2001), "Wind power development in Europe", *Proceedings of the IEEE*, Vol.89, pp. 1765–1782. DOI: https://doi.org/10.1109/5.975906
4. Lukas, M., Friedrich, K. (2017), "Wind energy statistics in Europe: onshore and offshore", *Wind Energy Management*, pp. 339–347. DOI:https://doi.org/10.1007/978-3-319-45659-1_36
5. Dubis, B., Dunn, J., Bełdycka-Bórawska, A., Jankowski, K., Bórawski, P. (2020), "Development of wind energy market in the European Union", *Renewable Energy*, 161, pp. 691–700. DOI:https://doi.org/10.1016/j.renene.2020.07.081
6. Viktor, P., Ali, B., Alzoubi, H., Jaszczur, M., Salman, H., Al-Musawi, T., Hassan, Q., Al-Jiboory, A., Sameen, A., Algburi, S. (2024), "The renewable energy role in the global energy transition", *Renewable Energy Focus*, 100545 p. DOI: https://doi.org/10.1016/j.ref.2024.100545
7. Alphan, H., Bilgili, M. (2022), "Global growth in offshore wind turbine technology", *Clean Technologies and Environmental Policy*, 24, pp. 2215–2227. DOI: https://doi.org/10.1007/s10098-022-02314-0
8. Lantz, E., Stehly, T., Shields, M., Cooperman, A., Kitzing, L., Beiter, P., Kikuchi, Y., Wiser, R., Berkhout, V., Telsnig, T. (2021), "Wind power costs driven by innovation and experience with further reductions on the horizon", *Wiley Interdisciplinary Reviews: Energy and Environment*, Vol. 10, Issue 5. DOI: https://doi.org/10.1002/wene.398

9. Baker, E., Beiter, P., Rand, J., Gilman, P., Seel, J., Wiser, R., Lantz, E. (2021), "Expert elicitation survey predicts 37% to 49% declines in wind energy costs by 2050", *Nature Energy*, 6, pp. 555–565. DOI: https://doi.org/10.1038/s41560-021-00810-z
10. Fan, Y., Peng, H., Zhang, H., Li, S., Shangguan, L. (2023), "Analysis of wind turbine equipment failure and intelligent operation and maintenance research", *Sustainability*, 15(10), 8333p. DOI: https://doi.org/10.3390/su15108333
11. Zhou, S., Wu, H., Du, Y., Peng, Y., Jing, X., Kwok, N. (2020), "Damage detection techniques for wind turbine blades: a review", *Mechanical Systems and Signal Processing*, Vol. 141, 106445 p. DOI: https://doi.org/10.1016/j.ymssp.2019.106445
12. Thomsen, K., Mishnaevsky, L. (2020), "Costs of repair of wind turbine blades: influence of technological aspects", *Wind Energy*, pp. 2247–2255. DOI: https://doi.org/10.1002/we.2552
13. Altice, B., Nazario, E., Davis, M., Shekaramiz, M., Moon, T., Masoum, M. (2024), "Anomaly detection on small wind turbine blades using deep learning algorithms", *Energies*, 17(5), 982 p. DOI: https://doi.org/10.3390/en17050982
14. Liu, Y., Zheng, Y., Shao, Z., Wei, T., Cui, T., Xu, R. (2024), "Defect detection of the surface of wind turbine blades combining attention mechanism", *Advanced Engineering Informatics*, 59, 102292 p. DOI: https://doi.org/10.1016/j.aei.2023.102292
15. Ogaili, A., Jaber, A., Hamzah, M. (2023), "A methodological approach for detecting multiple faults in wind turbine blades based on vibration signals and machine learning", *Curved and Layered Structures*, Vol. 10. DOI: https://doi.org/10.1515/cls-2022-0214
16. Dubchak, L., Sachenko, A., Bodyanskiy, Y., Wolff, C., Vasylkiv, N., Brukhanskyi, R., Kochan, V. (2024), "Adaptive neuro-fuzzy system for detection of wind turbine blade defects", *Energies*, 17(24), 6456 p. DOI: https://doi.org/10.3390/en17246456
17. Dubchak, L., Rusyn, B., Wolff, C., Ciszewski, T., Sachenko, A., Bodyanskiy, Y. (2026), "Hypersector-based method for real-time classification of wind turbine blade defects", *Energies*, 19, 442 p. DOI: https://doi.org/10.3390/en19020442
18. He, Y., Niu, X., Hao, C., Li, Y., Kang, L., Wang, Y. (2024), "An adaptive detection approach for multi-scale defects on wind turbine blade surface", *Mechanical Systems and Signal Processing*, 200, 111592 p. DOI: https://doi.org/10.1016/j.ymssp.2024.111592
19. Casal-Guisande, M., Cerqueiro-Pequeño, J., Bouza-Rodríguez, JB, Comesaña-Campos, A. (2023), "Integration of the Wang & Mendel algorithm into the application of fuzzy expert systems to intelligent clinical decision support systems", *Mathematics*, 11(1), 2469 p. DOI: https://doi.org/10.3390/math11112469
20. Chen, L., Su, W., Wu, M., Pedrycz, W., Hirota, K. (2020), "A fuzzy deep neural network with sparse autoencoder for emotional intention understanding in human–robot interaction", *IEEE Transactions on Fuzzy Systems*, 28(7), pp. 1252–1264. DOI: https://doi.org/10.1109/TFUZZ.2020.2966167
21. Talpur, N., Abdulkadir, SJ, Alhussian, H., Hasan, MH, Aziz, N., Bamhdi, A. (2022), "A comprehensive review of deep neuro-fuzzy system architectures and their optimization methods", *Neural Computing and Applications*, 34(1), pp. 1837–1875. DOI: https://doi.org/10.1007/s00521-021-06807-9
22. Vivchar, N. "Wang-Mendel", available at: https://github.com/NazarVivchar/wang-mendel/blob/59e6d3e9aec4533feeebd179cda53c7711a4dcdf/main.ipynb

*Received (Надійшла) 24.02.2026*
*Accepted for publication (Прийнята до друку) 21.05.2026*
*Publication date (Дата публікації) 27.06.2026*

*Відомості про авторів / About the Authors*

**Дубчак Леся Орестівна** – кандидат технічних наук, доцент, Західноукраїнський національний університет, завідувачка кафедри комп’ютерної інженерії, Тернопіль, Україна;

**Lesia Dubchak** – Candidate of Technical Sciences, Associate Professor, West Ukrainian National University, Head of the Computer Engineering Department, Ternopil, Ukraine;

e-mail: dlo@wunu.edu.ua

ORCID ID: https://orcid.org/0000-0003-3743-2432

Scopus Author ID: https://www.scopus.com/authid/detail.uri?authorId=56008186500

**Вівчар Назар Тарасович** – Західноукраїнський національний університет, здобувач вищої освіти ступеня доктора філософії кафедри комп’ютерної інженерії, Тернопіль, Україна;

**Nazar Vivchar** – West Ukrainian National University, Postgraduate Student of the Computer Engineering Department, Ternopil, Ukraine;

e-mail: nvivcharn@gmail.com

ORCID ID: https://orcid.org/0009-0003-8196-4978

Scopus Author ID: https://www.scopus.com/authid/detail.uri?authorId=59953891200

*Сучасний стан наукових досліджень та технологій в промисловості. 2026. № 2 (36)* *ISSN 2522-9818 (print)*
*Innovative technologies and scientific solutions for industries. 2026. No. 2 (36)* *ISSN 2524-2296 (online)*

**Савенко Олег Станіславович –** доктор технічних наук, професор, Хмельницький національний університет, професор кафедри комп'ютерної інженерії та інформаційних систем, Хмельницький, Україна;

**Oleg Savenko** – Doctor of Technical Sciences, Professor, Khmelnytskyi National University, Professor of the Computer Engineering and Information Systems Department, Khmelnytskyi, Ukraine;

e-mail: savenko_oleg_st@ukr.net

ORCID ID: https://orcid.org/0000-0002-4104-745X

Scopus Author ID: https://www.scopus.com/authid/detail.uri?authorId=54421023400

**Дериш Богдан Богданович –** Західноукраїнський національний університет, викладач кафедри комп'ютерної інженерії, Тернопіль, Україна;

**Bohdan Derysh** – West Ukrainian National University, Lecturer of the Computer Engineering Department, Ternopil, Ukraine;

e-mail: dbb@wunu.edu.ua

ORCID ID: https://orcid.org/0000-0002-7215-9032

**Заставний Олег Михайлович** – кандидат технічних наук, Західноукраїнський національний університет, старший викладач кафедри спеціалізованих комп'ютерних систем, Тернопіль, Україна;

**Oleg Zastavnyy** – Candidate of Technical Sciences, West Ukrainian National University, Senior Lecturer of the Specialized Computer Systems Department, Ternopil, Ukraine;

e-mail: o.zastavnyi@wunu.edu.ua

ORCID ID: https://orcid.org/0000-0001-8630-8791

Scopus Author ID: https://www.scopus.com/authid/detail.uri?authorId=8366871500

# ПРОГРАМНИЙ ЗАСІБ РЕАЛІЗАЦІЇ ІНТЕЛЕКТУАЛЬНОГО МЕТОДУ АНАЛІЗУ ДЕФЕКТІВ ЛОПАТЕЙ ВІТРОВИХ ТУРБІН З ОБМЕЖЕНОЮ ВИБІРКОЮ ДАНИХ

**Предметом дослідження** є методи й програмні засоби інтелектуальної діагностики дефектів лопатей вітрових турбін на основі нейронечітких моделей, здатних ефективно функціювати в умовах обмежених, нечітких або частково визначених вхідних даних. **Мета** – розроблення програмної реалізації та експериментальне дослідження нейронечіткої мережі Ванга–Менделя для класифікації дефектів лопатей вітрових турбін, а також обґрунтування доцільності її використання як основи інтелектуальної системи моніторингу технічного стану енергетичних об'єктів. **Завдання дослідження:** аналіз сучасних методів класифікації дефектів, розроблення архітектури нейронечіткої моделі, створення програмного комплексу для навчання й класифікації, реалізація алгоритму автоматичного формування бази нечітких правил, оптимізація параметрів функцій належності та проведення експериментальних досліджень для оцінювання ефективності запропонованого підходу. У роботі використано **методи** нечіткої логіки, нейронечітких систем, математичного моделювання, класифікації даних, оптимізації параметрів і програмної реалізації інтелектуальних систем мовою Python із застосуванням сучасних бібліотек оброблення даних і чисельних обчислень. Алгоритм Ванга–Менделя забезпечує автоматичне формування нечітких правил на основі навчальних даних і адаптацію параметрів моделі з метою підвищення точності класифікації. **Результати.** Розроблено інтелектуальний програмний комплекс для класифікації дефектів лопатей вітрових турбін, який забезпечує ефективне оброблення даних і підтримку прийняття рішень на основі нечітких правил. Експериментальні дослідження проведено на навчальній вибірці обсягом 160 записів і тестовій вибірці обсягом 64 записи, що описують імовірності наявності різних типів дефектів. **Висновки**: за результатами навчання досягнуто точності 91% на навчальній вибірці та 94% на тестовій вибірці за низького рівня середньоквадратичної похибки, що підтверджує високу ефективність, здатність до узагальнення та практичну придатність запропонованої моделі для використання в інтелектуальних системах моніторингу технічного стану вітрових турбін в умовах обмежених даних і ресурсів.

**Ключові слова:** вітрові турбіни; дефекти; нейронечіткі системи; алгоритм Ванга–Менделя; Python.

*Бібліографічні описи / Bibliographic descriptions*

Дубчак Л. О., Вівчар Н. Т., Савенко О. С., Дериш Б. Б., Заставний О. М. Програмний засіб реалізації інтелектуального методу аналізу дефектів лопатей вітрових турбін з обмеженою вибіркою даних. *Сучасний стан наукових досліджень та технологій в промисловості*. 2026. № 2 (36). С. 29–39. DOI: https://doi.org/10.30837/2522-9818.2026.2.029

Dubchak, L., Vivchar, N., Savenko, O., Derysh, B., Zastavnyy, O. (2026), "Software tool for implementing an intelligent method for analyzing wind turbine blade defects with limited dataset", *Innovative Technologies and Scientific Solutions for Industries*, No. 2 (36), P. 29–39. DOI: https://doi.org/10.30837/2522-9818.2026.2.029