Expert Systems with Applications 39 (2012) 6575–6584

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Pattern recognition with cerebellar model articulation controller and fractal features on partial discharges

Hung-Cheng Chen *, Feng-Chang Gu

Department of Electrical Engineering, National Chin-Yi University of Technology, Taichung 411, Taiwan

ARTICLE INFO

Keywords:
Cerebellar model articulation controller
Partial discharge
Pattern recognition
Fractal dimension
Lacunarity

ABSTRACT

This paper presents a new partial discharge (PD) pattern recognition method based on the cerebellar model articulation controller (CMAC). CMAC is an adaptive system by which defect types for partial discharge can be identified by referring to a table rather than by mathematical solution of simultaneous equations. CMAC maps input features of partial discharge into an input vector which is used to address a memory where the appropriate defect types are stored. Five types of defect models are well-designed on the base of investigation of many power apparatus failures. A PD detector is used to measure the raw three-dimension (3D) PD patterns, from which the fractal dimension, the lacunarity, and the mean discharges of phase windows are extracted as PD features. These critical features form the cluster domains of defect types. Using the characteristics of self-learning, association, and generalization, like the cerebellum of human being, the proposed CMAC-based pattern recognition scheme enables a powerful, straightforward, and efficient pattern recognition method. Moreover, the CMAC has the advantages of higher accuracy, shorter learning times, and noise tolerance, which are useful in recognizing the PD patterns of electrical apparatus. To demonstrate the effectiveness of the proposed method, comparative studies using a multilayer neural network (MNN) and K-means method are conducted on 200 sets of field-test PD patterns with high accuracy and high tolerance in noise interference.

1. Introduction

Partial discharges (PD) play an important role in the aging and breakdown of insulation system, and their presence may indicate defects which reduce lifetime and reliability of high-voltage apparatus. Partial discharge detectors have been widely applied to insulation diagnosis for high-voltage power apparatus, such as XLPE power cables, gas insulated switchgears, and power transformers (Leung & MacAlpine, 2002; Metwally, 2004; Naderi et al., 2008). The purpose of the insulation diagnosis for the power apparatus is to give system operators the information on dielectric deterioration degree of defect power equipments. And recently the information about these quantities has become obtainable, precise, and detailed. The main parameters of 3D PD patterns of the deterioration are widely used namely phase angle ϕ, discharge magnitude q, and number of discharge n. Each type of defect can be characterized by a specific 3D pattern. Therefore, the various defect types can be recognized and identified by the 3D patterns.

Various pattern recognition techniques, including fuzzy clustering (Galil, Sharkawy, Salama, & Bartnikas, 2005; Mazzetti et al., 2006) and neural network (NN) (Candela, Mirelli, & Schifani, 2000; Karthikeyan, Gopal, & Venkatesh, 2008; Karthikeyan, Gopal, & Vimala, 2005; Salama & Bartnikas, 2002), have been extensively used in PD recognition. The fuzzy approaches require human expertise and have been successfully applied to this field. A number of difficulties in acquiring knowledge and in maintaining the database are discussed. For example, the main advantage of the NN can directly acquire experience from the training data. However, the raw values of 3D patterns were used with the NN for PD recognition in previous studies (Salama & Bartnikas, 2002), the main drawbacks are that the structure of the NN has a great number of neurons with connections, and time consumptions in training. Another limitation of the NN approach is the inability to use linguistically descriptive output, because it is difficult to understand the content of network.

To overcome the drawbacks described above, a novel CMAC neural network based methodology is presented in this paper. Depending on the known defect types, the CMAC recognition structure is built first. By using the known fault patterns as the training data to train the CMAC, then the trained CMAC can be used to recognize the defect types of partial discharge. The characteristics of association and generalization enable the CMAC-based recognition scheme to become a powerful, straightforward, and accurate pattern recognition method. The proposed recognition system not only simplified the configuration, but also alleviated the dependency to experts' expertise. Furthermore, in order to improve the accuracy, the fractal dimension, the lacunarity, and the mean discharges of

* Corresponding author. Address: 35, Lane 215, Sec. 1, Chungshan Rd., Taiping, Taichung 411, Taiwan. Tel.: +886 423924505; fax: +886 423924419.
E-mail address: hcchen@ncut.edu.tw (H.-C. Chen).

0957-4174/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2011.12.044

phase windows are extracted from the raw values of 3D patterns as PD features for the proposed CMAC-based recognition system. The complex nature of the PD pattern shapes and the ability of fractal geometry to model complex shape (Jian, 2000; Zhao, Qiu, & Kuffel, 2002) are the main reasons which encouraged the authors to study the feasibility of fractal geometry for PD pattern interpretation. To demonstrate the effectiveness of the CMAC-based recognition method with fractal features enhancement, 200 sets of PD patterns are tested. The results show that the CMAC-based recognition method is considerably a practical solution.

2. Theory of the CMAC

The CMAC was first proposed by Albus in 1975 (Albus, 1975). It is used as a model of human memory performing reflexive processing to produce output signals of reflection immediately when cerebellum receives input signal. The similar input features will produce similar outputs. The CMAC is a kind of local learning feedforward neural network with simple architecture, quick learning convergence, and effective implementation. Although it is linear between different nerve cells, the mapping of the whole network is nonlinear. The basic aim of CMAC is to store data into overlapping regions in an associative manner such that the stored data can easily be recalled, although it uses less physical space (Liu & Wang, 2005).

2.1. Structure of the CMAC

In this section, to achieve highly accuracy and speed up the recognition, a CMAC recognition model is introduced. The structure of the CMAC is shown in Fig. 1, which contains (1) quantization, (2) segment address coding, (3) concatenation, and (4) summation of the fired memory address weights to obtain an output (Handeiman, Lane, & Gelfand, 1990). The mapping process must be satisfied that the similar inputs will excite the similar addresses. Therefore, if the input states are similar in input space, the output will have their corresponding sets of association cells overlap. The details are introduced in the following statements:

2.1.1. Quantization

As shown in Fig. 1 the input features are first through the quantization mapping to produces a quantization level output. The quantization output can be described as (Handeiman et al., 1990)

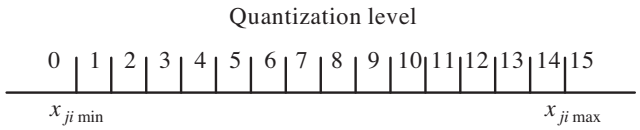

Fig. 2. Quantization mapping diagram.

$$q_{ji} = Q(x_{ji}, x_{ji\,\min}, x_{ji\,\max}, q_{i\,\max}), \quad j = 1, 2, \cdots, n \tag{1}$$

where n is the number of input features, i is defect type. The resolution of this quantization depends on the minimum and maximum, $x_{ji\,\min}$ and $x_{ji\,\max}$, and on the numbers of quantization levels $q_{i\,\max}$, which are extracted from input features. High resolution will have good generalization ability but needs with more memory size. The quantization level of each input features can be calculated as

$$q_{ji}(x_{ji}) = Ceil((x_{ji} - x_{ji\,\min})/((x_{ji\,\max} - x_{ji\,\min})/(q_{i\,\max} - 1))) \tag{2}$$

where $Ceil(x)$, a function rounding the elements of x to the nearest integers and extracting the elements of x_{ji} to the nearest positive integers. To simplify the quantization process, here we consider the $q_{i\,\max}$ as 16. That is, each input features will be quantized from 0 to 15. Fig. 2 shows the diagram of quantization mapping. For example, if $x_{ji\,\min} = 2.19$, $x_{ji\,\max} = 6.4$, and input $x_{ji} = 3.9$, we can calculate by Eq.(2) as $q_{ij} = Ceil((3.9 - 2.19)/(\{(6.4 - 2.19)/14\})) = 6$. The obtained the quantization levels of the input feature is 8. If $x_{ji} < x_{ij\,\min}$, we set the x_{ji} quantization level as 0. If $x_{ji} > x_{ji\,\max}$, then x_{ji} we set is 15. For each input features we will obtain corresponding quantization levels.

Then we code the quantization levels into binary code. The characteristic of CMAC NN is that similar inputs activate similar addresses. The fired memory address coding must satisfy this condition. Because using this method can reduce memory size and benefit for fired memory addresses.

2.1.2. Segment address mapping

When the quantization levels is finished, then performing V mapping shown in Table 1 outputs m segment addresses. m is the number of associated memory cells. Table 1 lists the mapping relation of the segment address and quantization levels, in which the quantization level $q_{i\,\max}$ and m are assumed to be 8 and 4, respectively. For example, the quantization level 6 be mapped into a group of segment addresses $[v_{1ji}, v_{2ji}, v_{3ji}, v_{4ji}] = [9,6,7,8]$.

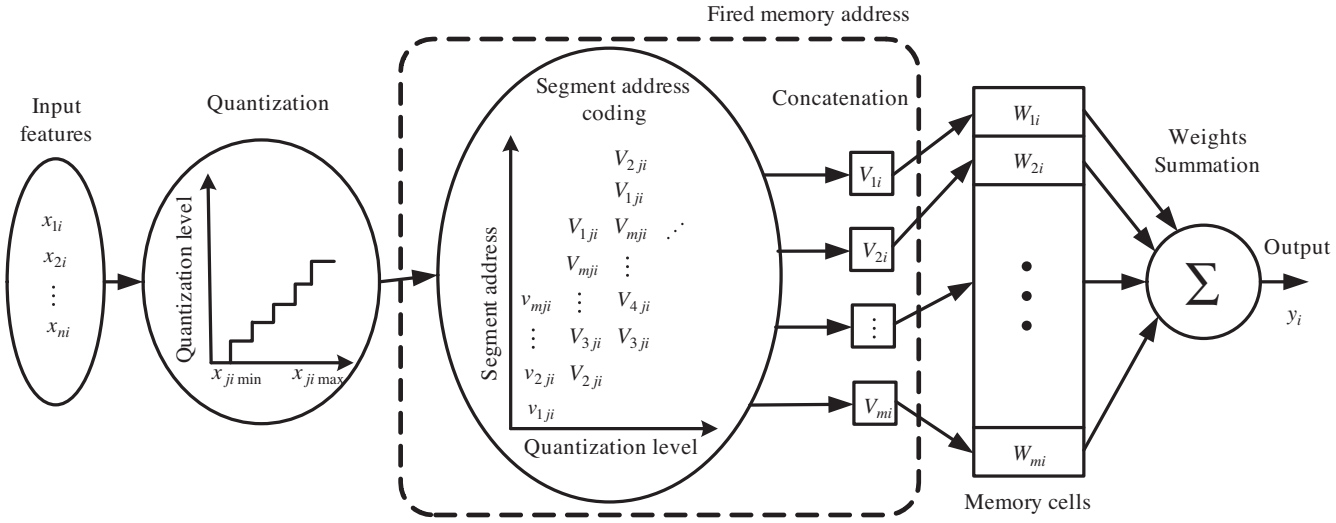

Fig. 1. Structure of the CMAC neural network.

Table 1
Mapping relation of quantization level and segment address.

Segment address	Quantization level								
11									v_{3ji}
10								v_{2ji}	v_{2ji}
9							v_{1ji}	v_{1ji}	v_{1ji}
8						v_{4ji}	v_{4ji}	v_{4ji}	v_{4ji}
7					v_{3ji}	v_{3ji}	v_{3ji}	v_{3ji}	
6				v_{2ji}	v_{2ji}	v_{2ji}	v_{2ji}		
5			v_{1ji}	v_{1ji}	v_{1ji}	v_{1ji}			
4	v_{4ji}	v_{4ji}	v_{4ji}	v_{4ji}					
3	v_{3ji}	v_{3ji}	v_{3ji}						
2	v_{2ji}	v_{2ji}							
1	v_{1ji}								
	1	2	3	4	5	6	7	8	

2.1.3. Concatenation and binary coding

Each input feature though segment address mapping obtains m segment addresses. The concatenation unit will concatenate these segment addresses as virtual addresses. The concatenation can be described as

$$V_{ji} = Concat(v_{j1i}, v_{j2i}, \cdots, v_{jni}) \quad for\ j = 1, 2, \cdots, m \tag{3}$$

The similar inputs activating the similar memory addresses is one of the most important characteristics of CMAC NN. The fired memory addresses coding has to satisfy this requirement. Utilizing binary representation will benefit the following fired memory addresses coding and memory size reduction. The minimum bit number to encode the segment addresses in binary representation is

$$N_{mb} = Ceil(\log_2(q_{i\max} + m - 1)) \tag{4}$$

In our study, for each type i the number of input features n is 12, the number of associated memory cells m is 10, and the number of quantization levels $q_{i\max}$ is 16. Therefore, the minimum bit number N_{mb} is 5 and the virtual addresses V_{ji} consist of 60 bits of binary codes.

2.1.4. Segmentation, fired addresses coding, and output mapping

Three bits are taken as a segment (group) from the 60 bits series in the virtual addresses which is expressed as 101 111...010 100 010 B for an illustration. From LSB to MSB, we extract the memory address as a_1 = 010B = 2, a_2 = 100B = 4, a_3 = 010B = 2, ... a_{19} = 111B = 7, a_{20} = 101B = 5. It implies that the number of the fired addresses A^* is 20. The features of the specific defect type are distributed and stored over the fired 20 memory addresses. Finally, adding the weights of the excited memory addresses, $w_1^2, w_2^4, w_3^2, \cdots, w_{19}^7, w_{20}^5$, will obtain the CMAC output. The output of CMAC can be expressed as

$$y_i = \sum_{j=1}^{m} \sum_{k=1}^{A^*} w_{kji}^{a_{kj}} \quad \text{for } i = 1, 2, \cdots, \ell \tag{5}$$

where a_{kj} are the fired memory addresses and ℓ is the number of defect types.

2.2. Training phase

In training phase, the patterns of type i are used to train the memory layer i, which is used to memorize the defect type i. An output indicated defect type can be obtained by the CMAC procedures of aforementioned Fig. 1, which is the feature input and the serious mapping, including quantization, concatenation, binary coding, segmentation, fired addresses coding, and summation of the fired memory addresses weights.

During the pattern training, if the output of the CMAC does not match the desired defect type, the weights at the fired addresses are updated using the following simple steepest-descent update rule

$$w_{kji(new)}^{a_{kj}} = w_{kji(old)}^{a_{kj}} + \beta \frac{y_d - y_i}{A^*} \quad \text{for} j = 1, 2, \cdots, m \quad \text{and} \quad k = 1, 2, \cdots, A^* \tag{6}$$

where $w_{kji(new)}^{a_{kj}}$ is the weight values after the weights tuning. a_{kj} denotes the fired memory addresses. β is the learning gain whose value is between interval [0,1] y_d = 1 is the desired output and y_i is the actual output.

The convergence is confirmed for a supervised learning system (Wong & Sideris, 1992). In this paper, we assume the memory size is large enough and the collision will not happen. In other word, the convergence is guaranteed. Assume that the memory layer i output 1 denotes the pattern belonging to type i, and the number of training patterns for defect type i is d. The performance index is defined as

$$E_i = \sum_{t=1}^{d} (y_i^t - 1)^2 \quad \text{for } i = 1, 2, \cdots, \ell \tag{7}$$

where the subscript i represents the ith defect type and the superscript t represents the tth training pattern. The training process will stop when $E_i < \varepsilon$ occurs, where ε is a small positive constant.

2.3. CMAC recognition method

The overall flow chart of the proposed recognition method based on CMAC NN is shown in Fig. 3. It can be simply described as follows

Step 1: Build the structure of CMAC fault recognition system. We define 12 input nodes, 5 output nodes, quantization level $q_{i\max} = 16$ (0–15), learning gain β = 0.9, and the number of fired addresses A^* = 20.
Step 2: Input the training patterns and perform quantization, segmentation, concatenation, and summation of the fired memory addresses weights to obtain an output.
Step 3: Calculate the actual output and compare with the desired output (y_d = 1), if the result does not match, then use Eq.(6) to update the weight values. We set 5 as the learning epoch.
Step 4: Evaluate the training performance. If $E_i < \varepsilon$, the training process is finished. Save the memory weights, others go to step 2.
Step 5: Carry out the pattern recognition. Input the patterns to be recognized, though quantization, segmentation, concatenation, and summation of the fired memory addresses weights which we have already trained to obtain an output.

Steps 1–4 construct an off-line mode. The training time may be shorter just few seconds or longer more than few hours depending on the data resolution, $q_{i\max}$, A^*, and the number of features. Fortunately, the off-line mode just only needs to run one time. Generally, the long training time will be better obtained better and more weights exacted, just like the learning mode of human being.

3. Extraction of PD feature

Fractals have been very successfully used to address the problem of modeling and to provide a description of naturally occurring phenomena and shapes, wherein conventional and existing mathematical methods were found to be inadequate (Chen, Keller, & Crownover, 1983; Satish & Zaeng, 1995). In recent years, this technique has attracted more attention for classification of textures

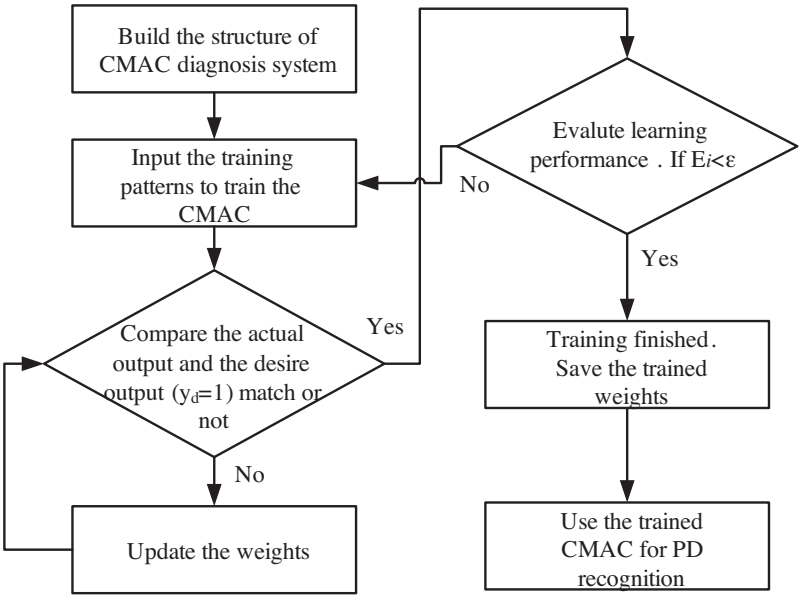

Fig. 3. The overall flow chart of the CMAC-based recognition method.

and objects present in images and natural scenes (Li, Du, & Sun, 2009; Tang, Tao, & Lam, 2002, and for modeling complex physical processes. In this theory, fractal dimensions are allowed to depict surface asperity of complicated geometric things. Therefore, it is possible to study complex objects with simplified formulas and fewer parameters (Voss, 1985). PD is also a natural phenomenon occurring in electrical insulation systems, which invariably contain tiny defects and non-uniformities, and gives rise to a variety of complex shapes and surfaces, both in a physical sense as well as in the shape of 3D PD patterns acquired using digital PD detector.

The fractal features, fractal dimension and lacunarity, and the mean discharges of phase windows are extracted to highlight the more detailed characteristics of the raw 3D PD patterns. The extracted features in this paper are introduced as follows:

3.1. Fractal dimension

While the definition of fractal dimension by self-similarity is straightforward, it is often difficult to estimate/compute for a given image data. However, a related measure of fractal dimension, the box dimension, can be more easily computed. In this work, the method suggested by Voss, and others in Chen et al. (1983) and Voss (1985), for the computation of fractal dimension D from the image data has been followed. Let $p(m,L)$ define the probability that there are m points within a box of size L (i.e. cube of side L), which is centered about a point on the image surface. $P(m,L)$ is normalized, as below, for all L.

$$\sum_{m=1}^{N} p(m,L) = 1 \tag{8}$$

where N is the number of possible points within the box. Let S be the number of image points (i.e. pixels in an image). If one overlays the image with boxes of side L, then the number of boxes with m points inside the box is $(S/m)p(m,L)$. Therefore, the expected total number of boxes needed to cover the whole image (Voss, 1985) is

$$N(L) = \sum_{m=1}^{N} \frac{S}{m} p(m,L) = S \sum_{m=1}^{N} \frac{1}{m} p(m,L). \tag{9}$$

This value is also proportional to L^{-D} and the box dimension can be estimated by calculating $p(m,L)$ and $N(L)$ for various values of L, and by doing a least square fit on $[\log(L), -\log(N(L))]$. To estimate $p(m,L)$, one must center the cube of size L around an image point and count the number of neighboring points m that fall within the cube. Accumulating the occurrences of each number of neighboring points over the image gives the frequency of occurrence of m. This is normalized to obtain $p(m,L)$. Values of L are chosen to be odd to simplify the centering process. Also, the centering and counting activity is restricted to pixels having all their neighbors inside the image. This will obviously leave out image portions of width = $(L-1)/2$ on the borders. This reduced image is then considered for the counting process. Consequently, large values of L results in increased image areas from being excluded during the counting process, thereby increasing uncertainty about counts near border areas of the image. This is one of the sources of errors for the estimation of $p(m,L)$ and thereby D. Additionally, the computation time grows with the L value. Hence, L = 3, 5, 7, and 11 are chosen for this work.

3.2. Lacunarity

Theoretically, ideal fractal could confirm to statistical similarity for all scales. In other words, fractal dimensions are independent scales. However, it has been observed that fractal dimension alone is insufficient for purposes of discrimination, since two differently appearing surfaces could have the same value of D. To overcome this, Mandelbrot introduced the term called lacunarity Λ, which quantifies the denseness of an image surface. Many definitions of this term have been proposed and the basic idea in all these is to quantify the 'gaps or lacunae' present in a given surface. One of the useful definitions of this term as suggested by Mandelbrot (1983, chap. 5) is

$$M(L) = \sum_{m=1}^{N} mp(m,L) \tag{10}$$

and

$$M^2(L) = \sum_{m=1}^{N} m^2 p(m,L) \tag{11}$$

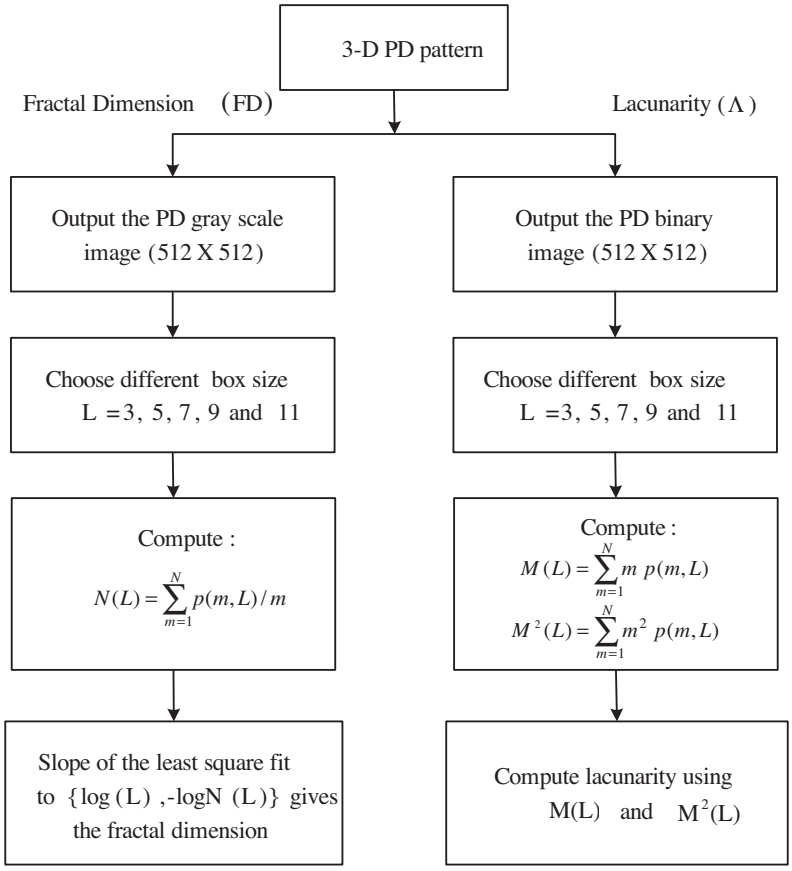

Fig. 4. Procedure for computing fractal dimension and lacunarity.

where N is the numbers of point in the data set of size L, the lacunarity becomes

$$\Lambda(L)=\frac{M^2(L)-[M(L)]^2}{[M(L)]^2} \tag{12}$$

Fig. 4 shows the procedure for extracting fractal features. In fractal dimension extraction the first step is to transfer PD pattern to a 512 × 512 matrix. $N(L)$ is then obtained by using different box size L. Finally, the fractal dimension can be obtained by fitting the data [log$L$, −log($N(L)$)].

In the lacunarity extraction the first step is to transfer PD pattern to a binary image 512 × 512 matrix and then to choose different box size L. We choose L = 3, the best result for the lacunarity, to compute the $M(L)$ and $M^2(L)$. Finally, the lacunarity can be computed by Eq.(12).

3.3. Mean values of discharges

The detailed features extraction process of mean values of discharges is shown in Fig. 5. A 3D PD pattern is divided into 10 phase windows whose width is set to 36°. The mean discharge is calculated in every phase window. We will obtain 10 mean discharge parameters on the whole 360° phase angles. If each phase window is further divided into $n \times m$ equal sections, the mean value of each phase window can be calculated by

$$v_i=\frac{\sum_{j=1}^{m}\sum_{k=1}^{n} q_j n_{jk}}{\sum_{j=1}^{m}\sum_{k=1}^{n} n_{jk}} \quad for \quad i=1,2,\cdots,10 \tag{13}$$

where q_j is the discharge magnitude in jth row. n_{jk} is the discharge number in jkth section.

4. PD recognition system design

The block diagram of the designed PD recognition system is shown in Fig. 6. It consists of three main parts: well-designed defect models, measurement system, and CMAC-based recognition method. The details of the three main parts are introduced in subsequent paragraphs.

4.1. Defect model

According to the fact that gap discharge and surface discharge are more likely to occur in high-voltage power equipments, five types of relevant models are well-designed on the base of investigation of many power equipment failures to outline the features of PD.

T1: Plane to plane model. A 9 mm-in-thickness epoxy with a 3 mm-in-diameter cylindrical cavity is inserted.

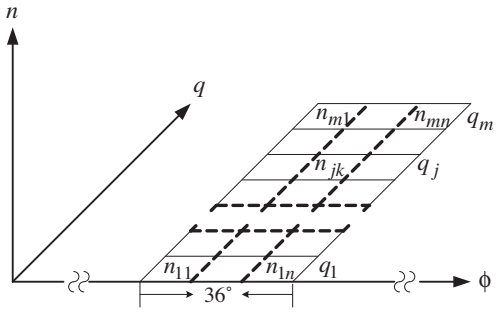

Fig. 5. Illustration for the calculation of mean values of charges.

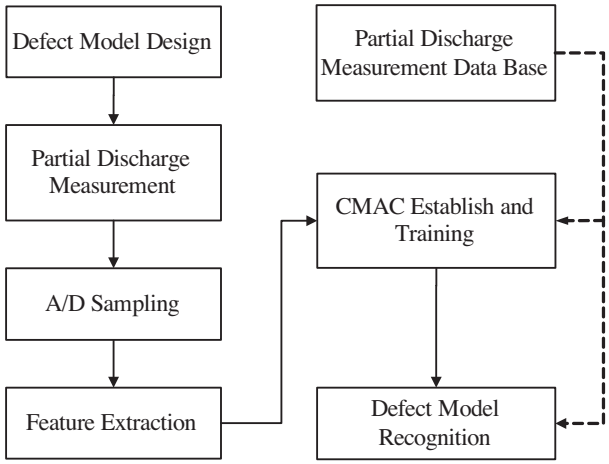

Fig. 6. The block diagram of the designed PD recognition system.

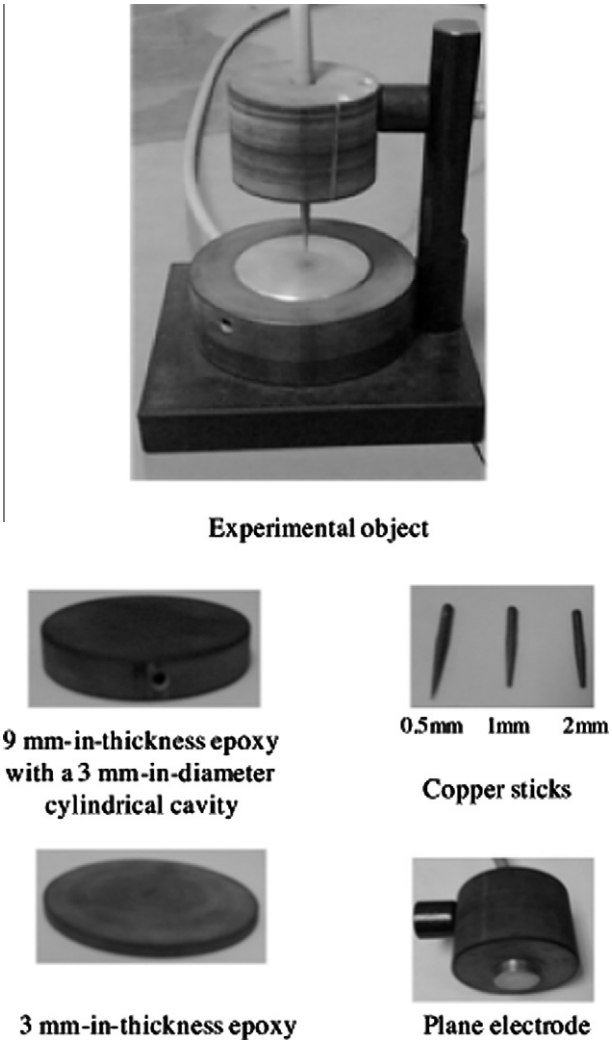

Fig. 7. The practical specimens of the defect model.

T2: Plane to plane model. A 3 mm-in-thickness epoxy is inserted.
T3: Needle to plane model. A 4 mm-in-diameter copper stick is lathed at one end to a 30° 0.5 mm-in-diameter cone needle which is 6 cm away from the plane.
T4: Needle to plane model. A 4 mm-in-diameter copper stick is lathed at one end to a 30° 1 mm-in-diameter cone needle which is 6 cm away from the plane.
T5: Needle to plane model. A 4 mm-in-diameter copper stick is lathed at one end to a 30° 2 mm-in-diameter cone needle which is 6 cm away from the plane.

Both the plane and the needle are made of copper. The practical specimens of the defect models are shown in Fig. 7.

4.2. Measurement system

The structure of the measuring system is shown in Fig. 8. The autotransformer is used to slowly rise the output voltage of the transformer to 5.4 kV as the testing voltage on defect model. The detector LDP-5 equipped with a capacitive sensor measures the PD electrical signal generated by the defect model. The PD signal is converted into a computer by NI DAQ card (PCI-6110) for further analysis. For each type of defect model, the measurement is conducted for 40 times. The sampling rate of the PCI-6110 DAQ card is set to 2 M/s and data acquisition duration per measurement is 24 cycles (60 Hz). The acquired PD signal is transferred into a 3D pattern. The features of the 3D pattern are extracted and used as the input parameters of the recognition system based on ENN recognition method.

A man–machine interface for the PD measurement is designed using a LabVIEW software. Analyzing the signals through the LabVIEW can obtain the instant values of PD signals and the comparison with the testing voltage (60 Hz) in real time. The designed man–machine interface for PD measurement is shown in Fig. 9. The red and green curves indicate the PD signal and the testing voltage waveform, respectively.

We use PD detector LDP-5 equipped with a capacitive sensor to measure the PD signal of the defect models. A typical PD impulse detected and manipulated by LDP-5 is shown in Fig. 10. The rise time is faster than the fall time. The frequency range of the impulse is 30–40 kHz.

5. Experiment results and discussions

The proposed CMAC-based recognition method has been implemented according to the measured PD pattern on the defect models. There are a total of 200 sets of measurement data associated with the five types of defect models. Some important experiment results are shown as follows:

5.1. 3D PD patterns

The typical 3D PD patterns transferred from the measured PD signals for each defect model are shown in Fig. 11. The main parameters of the 3D PD patterns are phase angle φ, discharge magnitude q, and the numbers of discharge n. Observing the results, the number of discharge in type T1 is less then that in type T2, but distribution in type T1 is wider than that in type T2. We can observe easily that the numbers of discharge in types T3, T4, and T5 are greater than that in types T1 and T2. A large number of the discharges exist when the needle-tip is thinner. The discharges of needle to plane model almost happen in positive period. Sometimes types T3 and T4 having discharges happen in negative period, in which the number of discharge is small, but the discharge magnitude is large. According to the 3D PD pattern in each defect model, we can find some different features between each defect models.

5.2. Feature extraction

The features of a total of 200 sets of 3D PD patterns are extracted and used as the input parameters of the recognition system

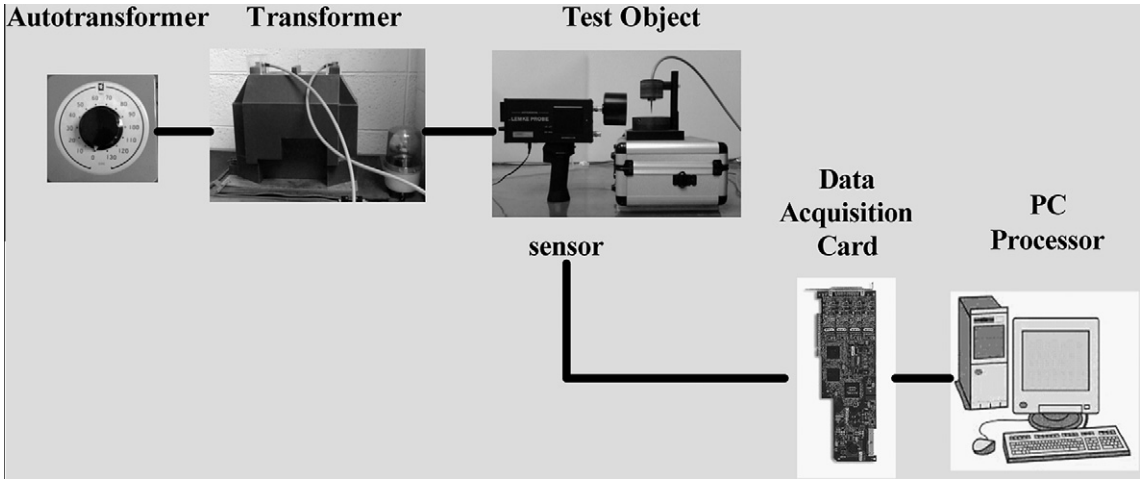

Fig. 8. The structure of the measurement system.

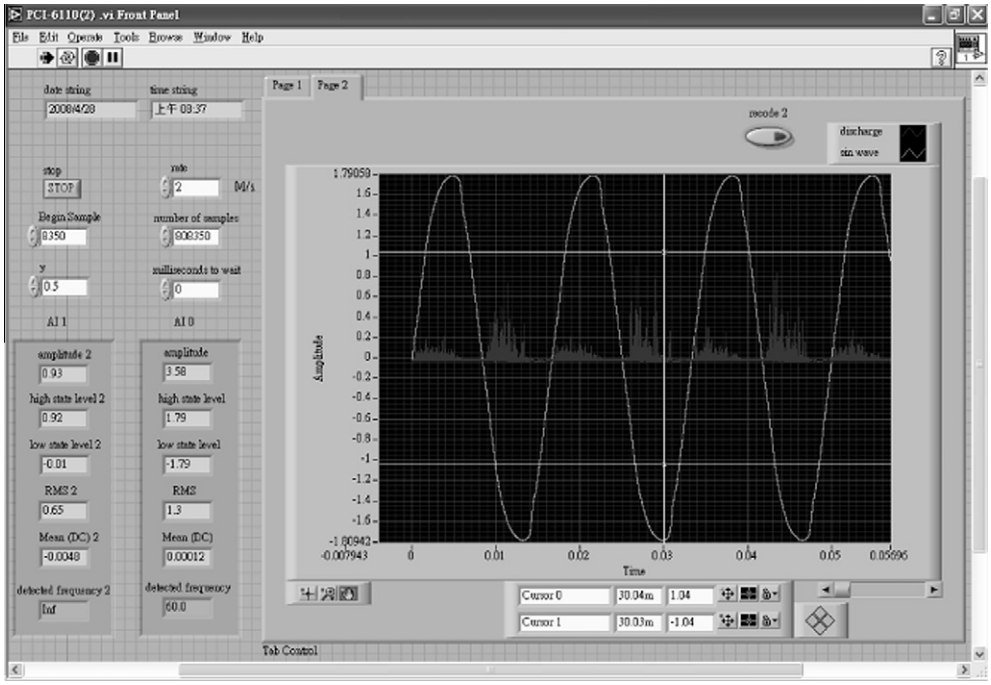

Fig. 9. The designed man–machine interface for PD measurement.

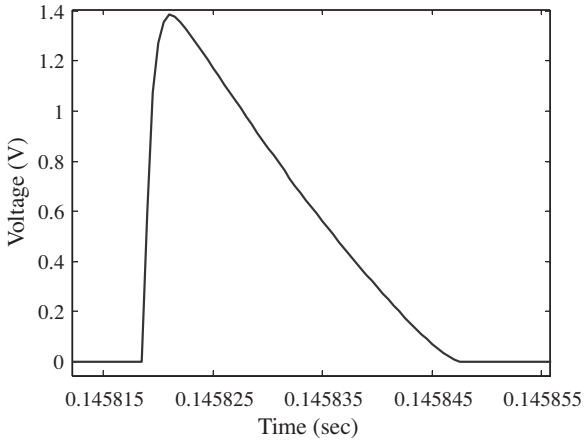

Fig. 10. The PD current impulse.

base on CMAC recognition method. Two features, the fractal dimension and the lacunarity, are calculated based on the fractal theory. The distribution of the fractal dimensions and the lacunarities of all 3D PD patterns are shown in Fig. 12. It is obvious that features belonging to a particular defect type gather together. According to the two fractal features, types T1 and T2 can be easily classified. However, the distributions of types T3, T4, and T5 overlap somewhere, which causes inaccurate classification. Therefore we take the mean values of discharge associated with phase windows as additional features. The mean discharges of all type of defect models associated with phase windows are shown in Fig. 13. We can find that the distribution of T1 mean discharge magnitude is wider than the others. T4 mean discharge magnitudes in phase windows six to nine have the maximum discharge than the others. In Fig. 13, we can find that many obvious differences exist in each defect type model.

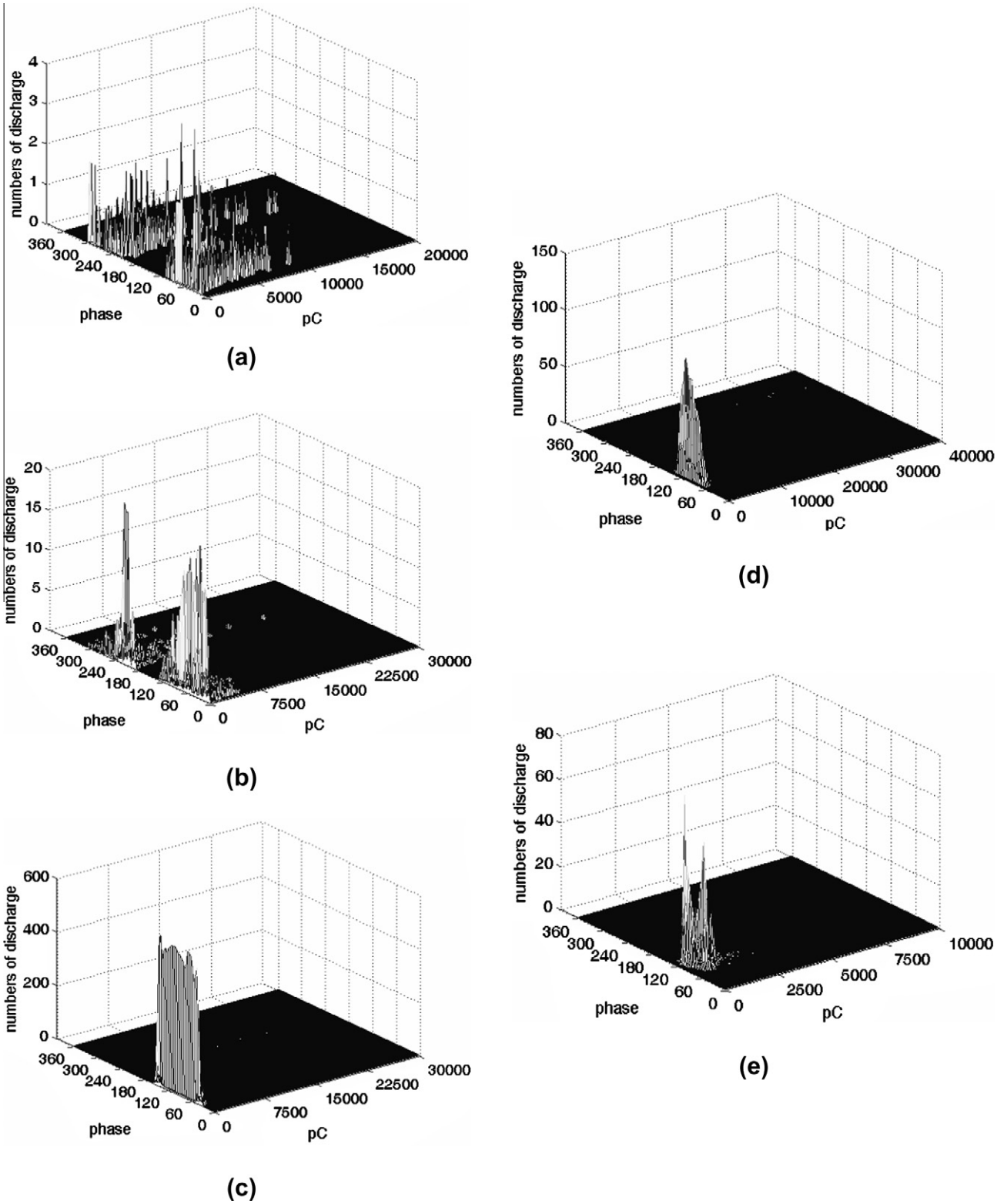

Fig. 11. Five typical defect type of 3D PD patterns. (a) type T1 (b) type T2 (c) type T3 (d) type T4 (e) type T5.

5.3. Performance evaluation of the proposed CMAC method

Table 2 shows the performance of the CMAC-based recognition method compared with the MNN and K-means. It can found that the recognition speed of K-means is the fastest one than any others, but the recognition accuracy rate is the lowest. It should be noted that the structure of the CMAC is simpler than that of the MNN because of simple mapping and calculation. Moreover, the CMAC-based recognition method also permits fast adaptive processing for a large amount of training data. The CMAC method not only takes experts' experience from learning, but also produces meaningful output after learning, because the optimal classified

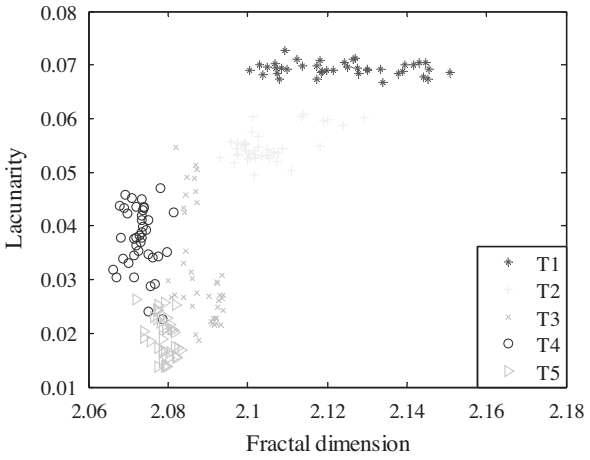

Fig. 12. Distribution of fractal features of all model.

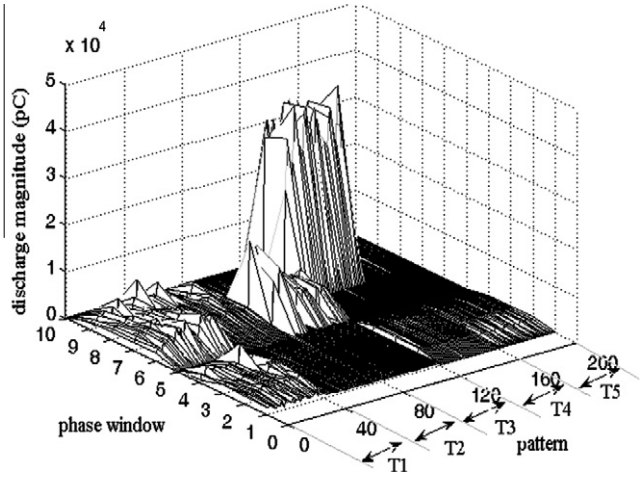

Fig. 13. Mean discharge magnitude in phase windows.

boundaries of the features are clearly determined. It can also be observed from Table 2 that the CMAC has shorter recognition time than MNN. The learning epochs of the CMAC-based recognition method we set are 10 epochs. It totally spends 0.43s of CPU time. Compared with MNN, the MNN needs 1000 learning epochs so that the output can arrive at the target, because the structure and the calculation of the MNN are more complicated. It will totally spend 105 CPU time (sec). Although the PD recognition system is trained offline, the training time may not be critical to be evaluated. However, if we want to implement the PD recognition system in a microcomputer for a real-time PD detecting device or portable instrument, the performance of training would be important.

5.4. Recognition accuracy of the proposed CMAC recognition method

The PD signal would unavoidably contain some noise. The sources of noise may be generated from the PD detector, the environmental electromagnetic, or the human mistakes, etc. To take the noise into account, 200 sets of testing data are created by adding the random uniformly distributed noises from ±10% to ±30%. The recognition accuracy rates with different amounts of noise added are given in Table 3. To demonstrate the effectiveness of the proposed method, comparative studies using a MNN with three layers and 12-12-5 neurons and K-means algorithm are all conducted on the same testing data. In the case of no noise added,

Table 2
Performance comparison of the CMAC method.

Compared items	Recognition method		
	CMAC	MNN	K-means
Structure	12-5	12-12-5	12-5
No. of connection	–[a]	204	–[a]
Learning epochs	10	1000	–[b]
CPU time (sec)	0.43	105	0.03

[a] Connections between input and output are unnecessary.
[b] Learning epochs are unnecessary.

Table 3
Accuracy rates of PD recognition.

Noise (%)	Average recognition rates[a] (%)		
	CMAC (%)	MNN (%)	K-means (%)
0	96.0	100	82.0
±10	96.4	98.5	73.2
±20	95.6	89.9	69.8
±30	94.0	83.8	67.5

[a] Average of 10 random trials.

the recognition accuracy of the MNN arrives at 100% which is slightly better than that of CMAC at 96%. However, the accuracy of CMAC recognition even is 94% in the case of ±30% noise added. The accuracies of the MNN and the K-means algorithm are only 83.8% and 67.5% in the same condition, respectively. It shows that the proposed method has pretty high recognition accuracy and good tolerance in noise interference.

6. Conclusions

This paper presents a new PD recognition method based on the fractal features and the CMAC. The fractal features are used to highlight the more characteristics of the raw 3D PD patterns in detail. Using training patterns to train CMAC like the brain of human being, each defect type features is distributed and memorized on an assigned memory layer. In the experiment, the recognition rates of the proposed method are quite high up to 94% in extreme noise of ±30%. The experimental results indicate that this method is able to implement an efficient classification with a very high recognition rate. Compared with the MNN method, the proposed method needs learning processes but it takes quite few training times. In addition, the calculation of the proposed recognition algorithm is fast and very simple. This new method merits more attention to be considered as a useful tool in PD recognition problems.

Acknowledgment

The research was supported by the National Science Council of the Republic of China, under Grant No. NSC99-2213-E-167-030-MY3.

References

Albus, J. S. (1975). A new approach to manipulator control: The cerebellar model articulation controller (CMAC). *Transaction of ASME Journal of Dynamic Systems, Measurement, and Control, 97*, 220–227.

Candela, R., Mirelli, G., & Schifani, R. (2000). PD recognition by means of statistical and fractal parameters and a neural network. *IEEE Transaction on Dielectrics and Electrical Insulation, 7*(1), 87–94.

Chen, S. S., Keller, J. M., & Crownover, R. M. (1983). On the calculation of fractal features from images. *IEEE Transaction on Pattern Analysis and Machine Intelligence, 15*, 1087–1090.

Galil, T. K. A., Sharkawy, R. M., Salama, M. M. A., & Bartnikas, R. (2005). Partial discharge pattern classification using the fuzzy decision tree approach. *IEEE Transaction on Instrumentation and Measurement, 54*(6), 2258–2263.

Handeiman, D. A., Lane, S. H., & Gelfand, J. J. (1990). Integrating neural networks and knowledge-based systems for intelligent robotic control. *IEEE Control System Magazine, 10*, 77–87.

Jian, L., et al. (2000). Pattern recognition of partial discharge with fractal analysis to characteristic spectrum. In Proceedings of the 6th International Conference on Properties and Application of Dielectric Materials, pp. 21–26.

Karthikeyan, B., Gopal, S., & Venkatesh, S. (2008). Partial discharge pattern classification using composite versions of probabilistic neural network inference engine. *Expert Systems with Applications, 34*(3), 1938–1947.

Karthikeyan, B., Gopal, S., & Vimala, M. (2005). Conception of complex probabilistic neural network system for classification of partial discharge patterns using multifarious inputs. *Expert Systems with Applications, 29*(4), 953–963.

Leung, Y. C., & MacAlpine, J. M. K. (2002). Initial experience with the partial discharge monitoring of high-voltage motors. *International Journal of Electrical Power and Energy Systems, 61*(1), 33–40.

Li, J., Du, Q., & Sun, C. X. (2009). An improved box-counting method for image fractal dimension estimation. *Pattern Recognition, 42*(11), 2460–2469.

Liu, J. C., & Wang, Z. (2005). CMAC-based fuzzy controller for strip flatness pattern recognition. *Journal of Northeastern University (Natural Science), 26*(8), 718–721.

Mandelbrot, B. B. (1983). *Fractal geometry of nature*. New York: Freeman.

Mazzetti, C. et al. (2006). Partial discharge pattern recognition by neural-fuzzy networks in heat-shrinkable joint and terminations of XLPE insulated distribution cables. *IEEE Transaction on Power Delivery, 21*(3), 1035–1044.

Metwally, I. A. (2004). Status review on partial discharge measurement techniques in gas-insulated switchgear/lines. *International Journal of Electrical Power and Energy Systems, 69*(1), 25–36.

Naderi, M. S. et al. (2008). Application of wavelet analysis to the determination of partial discharge location in multiple-α transformer windings. *International Journal of Electrical Power and Energy Systems, 78*(2), 202–208.

Salama, M. M. A., & Bartnikas, R. (2002). Determination of neural network topology for partial discharge pulse pattern recognition. *IEEE Transaction on Neural Networks, 13*(2), 446–456.

Satish, L., & Zaeng, W. S. (1995). Can fractal features be used for recognizing 3-D partial discharge patterns? *IEEE Transaction on Dielectrics and Electrical Insulation, 2*, 352–359.

Tang, Y. Y., Tao, Y., & Lam, E. C. M. (2002). New method for feature extraction based on fractal behavior. *Pattern Recognition, 35*(5), 1071–1081.

Voss, R. F. (1985). *Random fractal: Characterization and measurement*. New York: Plenum Press. Chapter 4.

Wong, Y. F., & Sideris, A. (1992). Learning convergence in the cerebellar model articulation controller. *IEEE Transaction on Neural Networks, 3*(1), 115–121.

Zhao, Z., Qiu, Y., & Kuffel, E. (2002). Application of fractal to PD signal recognition. In Proceedings of IEEE International Symposium on Electrical Insulation, pp. 523–526).