Sensors and Materials, Vol. 36, No. 2 (2024) 639–653
MYU Tokyo

S & M 3552

Automated Detection of Heart Arrhythmia Signals by Using a Convolutional Takagi–Sugeno–Kang-type Fuzzy Neural Network

Cheng-Jian Lin,[1*] Han Cheng,[1] and Chun-Lung Chang[2]

[1]Department of Computer Science & Information Engineering, National Chin-Yi University of Technology, Taichung 411, Taiwan
[2]Department of Artificial Intelligence & Computer Engineering, National Chin-Yi University of Technology, Taichung 411, Taiwan

(Received June 10, 2023; accepted January 29, 2024)

Keywords: arrhythmia detection, electrocardiogram, TSK-type fuzzy neural network, uniform experimental design

In clinical practice, electrocardiography is used to diagnose cardiac abnormalities. Because of the extended time required to monitor electrocardiographic signals, the necessity of interpretation by physicians, and the vulnerability of electrocardiographic signals to noise interference, electrocardiography is laborious and places a heavy burden on healthcare providers. Therefore, in this paper, a convolutional Takagi–Sugeno–Kang (TSK)-type fuzzy neural network (CTFNN) is proposed to address the challenges of arrhythmia signal classification. The proposed CTFNN is divided into three parts, namely, a convolutional layer, a feature fusion layer, and a TSK fuzzy neural network. The TSK fuzzy neural network is used to replace the fully connected neural network, thereby reducing the number of parameters and enabling the model to mimic the human brain when classifying signals. In addition, because the parameters of the CTFNN are difficult to determine, the uniform experimental design method, which requires only a small number of experiments, is used to determine the optimal parameter combination. The proposed model was tested using the Massachusetts Institute of Technology-Beth Israel Hospital (MIT-BIH) arrhythmia database, which contains 1000 records belonging to 17 categories. Each record has a duration of 10 s and contains 3600 sampling points. According to our experimental results, the accuracy, recall, precision, and F1-score of the CTFNN for long-term signals were 97.33, 97.96, 96.00, and 96.97%, respectively. In addition, the number of parameters for the proposed model was only 558,728, which was less than that for LeNet (i.e., 1734501).

1. Introduction

According to the World Health Organization, heart disease is the leading cause of death worldwide. Since 2000, among the deaths caused by various factors, the deaths caused by heart disease have exhibited the highest increase. Heart disease kills 17.9 million people worldwide

*Corresponding author: e-mail: cjlin@ncut.edu.tw
https://doi.org/10.18494/SAM4682

ISSN 0914-4935

every year. The typical resting heart rate for an adult is 60–100 beats per minute. Arrhythmia is an alteration in the rhythm of the heartbeat that might result from the delay or blockage of the electrical signal that controls the heartbeat or from the failure of the sinoatrial node to function normally. Although most arrhythmias are benign and not life-threatening, malignant arrhythmias are likely to be a manifestation of heart disease or stroke. Ventricular tachycardia and fibrillation are extremely dangerous arrhythmias that can lead to the sudden death of the patient. At present, the clinical diagnosis of arrhythmia involves listening to heartbeats and chest palpitations, general 12-lead electrocardiography (ECG), and electrophysiology. ECG is preferred over electrophysiology because the latter is invasive. Occasionally, a patient's heart rate may be monitored continuously for periods of 24 h to 14 days by using a small battery-powered electrocardiogram device attached to the body.

Researchers have applied machine learning approaches to medical imaging and biomedical monitoring for diagnostic analysis. Kumari *et al.*[1] used discrete wavelet transform to extract signal features from ECG samples. Support vector machine[2] was used to classify normal sinus rhythm, congestive heart failure, and cardiac arrhythmia. Zou *et al.*[3] used random forest to classify ECG signals from a heart rate sensor. They proposed using discriminative labels to extract long-period signal features, which can provide the contextual information of specific heartbeats to improve the performance of heartbeat classifiers. Yang and Wei[4] proposed a visual morphological model for analyzing the Q-, R-, and S-wave (QRS) complexes according to ECG morphology and used the K-nearest neighbor algorithm to classify six heartbeat types. Conventional machine learning involves using large data sets to identify patterns and make predictions about future data points. The aforementioned methods require the manual selection of signal features, which is labor-intensive. For convenience, deep learning methods have been developed to allow models to extract these features automatically.

Several researchers have used deep learning methods for the identification of arrhythmia. Convolutional neural networks (CNNs) are widely used network architectures. Wu *et al.*[5] aimed to analyze short-term, single-lead ECG signals by time–frequency transformation to obtain the signal distribution matrix. They used a CNN to classify 12 arrhythmias and evaluated the performance of three time–frequency transformation methods in this classification. Mahmud *et al.*[6] proposed a novel deep CNN architecture that uses pointwise–temporal–pointwise convolutional structural units with different temporal orders. They used residuals combined with features at different time points for arrhythmia classification. Zhang *et al.*[7] proposed the ECGNet architecture to improve the diagnostic accuracy of premature ventricular complexes in small data sets with imbalanced samples. They used convolution kernels of different sizes and a new loss function to improve the classification performance of ECGNet on ECG images. Because most CNN models use multiple convolutional and pooling layers to extract feature weights or reduce dimensions before finally performing classification through fully connected layer calculations, two difficulties occur. First, the fully connected layer is similar to a black box, and the weighted value has no physical meaning. Second, users often determine model parameters through trial and error, which leads to an increase in experimental cost.

Fuzzy logic mimics human brain reasoning and is often used to describe the correlation between input and output. Zadeh[8] proposed the concept of fuzzy sets, which is different from

the binary system commonly used in general mathematics. The binary system, which uses only zeros and ones, does not allow any ambiguity. The fuzzy set calculates the membership degree through the membership function and quantifies and infers the fuzzy set and rules. Jang[9] combined fuzzy logic and a neural network approach to develop an adaptive network-based fuzzy inference system architecture that can automatically determine the parameters of fuzzy if–then rules through gradient learning. Wu *et al.*[10] used a three-dimensional CNN model to extract features and adopted a fuzzy min–max neural network as a classifier for open action recognition. Boreiri *et al.*[11] used the fuzzy two-stage color segmentation algorithm for image preprocessing and proposed the CNN of the Takagi–Sugeno–Kang (TSK)-type fuzzy model to detect acute lymphoblastic leukemia. Du *et al.*[12] proposed a TSK-type convolutional recurrent fuzzy network to predict driver fatigue in EEG signals.

Because CNN model parameters are often determined through trial and error, the cost of experiments is high. In engineering, the Taguchi[13–15] and uniform experimental design (UED)[16–18] methods are often used to optimize experimental parameters and increase efficiency. In this study, UED was applied for parameter selection.

In this paper, we propose a convolutional TSK-type fuzzy neural network (CTFNN) for arrhythmia signal classification. The main contributions of this study are as follows:

- A CTFNN, which combines a CNN and TSK-type fuzzy neural network, is proposed to address the shortcomings of deep learning networks in arrhythmia classification.
- A feature fusion layer with global max pooling is proposed for feature extraction.
- The UED method is applied to optimize parameter selection in the proposed CTFNN model.
- Experimental results indicated that the accuracy, recall, precision, and F1-score of the proposed CTFNN for long-term ECG signals were 97.33, 97.96, 96.00, and 96.97%, respectively.
- The proposed CTFNN model had only 558,728 parameters, whereas the LeNet model had 1734501 parameters.

The rest of this paper is organized as follows. Section 2 provides a detailed introduction of the UED method and CTFNN model. Section 3 presents the experimental results obtained using the CTFNN on the Massachusetts Institute of Technology-Beth Israel Hospital (MIT-BIH) arrhythmia data set. Section 4 provides the conclusions of this study and recommendations for future research.

2. Materials and Methods

In this paper, we propose an arrhythmia classification system that combines UED and a CTFNN and uses long-term ECG signals. UED, which is efficient, is used to determine the parameters of the network, and a TSK-type fuzzy neural network (TFNN) is used to replace the fully connected network for reducing the number of parameters. A flow chart of the proposed system architecture is shown in Fig. 1.

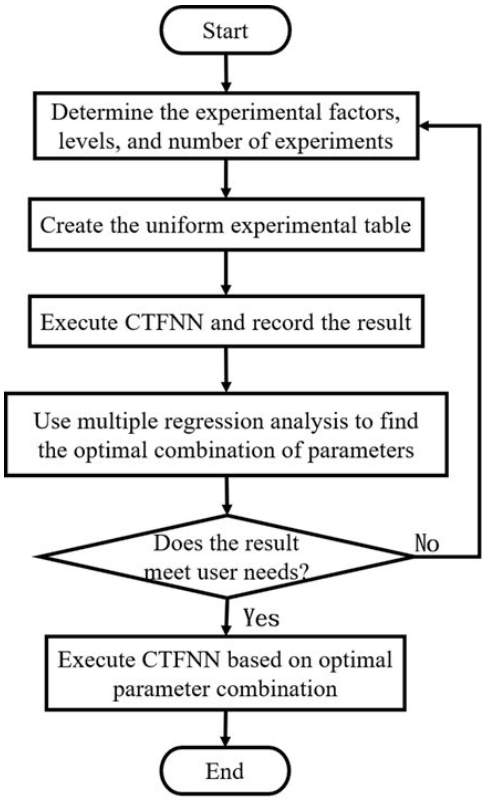


Fig. 1. System architecture.

2.1 Uniform experimental design

UED is an application of the quasi-Monte Carlo method based on the theory of numbers.[16] UED only considers the uniform dispersion of test points within the test range, and each test point is representative. UED requires fewer experiments than does the Taguchi method when the same numbers of factors and levels are used, which makes it more suitable than the Taguchi method for multifactor and multilevel models or in situations where the model is completely unknown. However, if the number of experiments is insufficient, the network model is unable to provide accurate results. The UED table is usually represented by $U_n(q^s)$, where n represents the number of experiments, q is the number of levels, and s is the number of factors.

First, the problem to be solved is defined. Second, the experimental factors and their levels and the number of experiments are determined. The number of experiments must be at least twice the number of factors; otherwise, an effective model might not be established. An initial UED table is designed according to the selected factors and levels. On the basis of the congruence theorem, the structure of the UED table is calculated using its jth row and ith column as

$$u_{ij} = \mathrm{i} \times j \left[mod\, n \right] \cdot \tag{1}$$

The good lattice point method evenly distributes representative points to create a UED table. Third, the established uniform table is used in conjunction with its UED table. In the table used, the number of rows indicates that the row has a higher uniformity and a smaller deviation. The centered L2-discrepancy is used to assess whether UED tables have uniform dispersion. Fourth, experiments are conducted according to the configured UED table, and the experimental results are recorded. Finally, multiple regression analysis is conducted using the following formula:

$$\varepsilon = Y - [\alpha_0 + \sum_{i=1}^{f} \alpha_{1i}\beta_i + \sum_{i=1}^{f} \alpha_{2i}\beta_i^2 + \sum_{i=1}^{f} \alpha_{3i}\beta_i^3 + \sum_{i=1}^{f-1} \sum_{j=i+1}^{f} \alpha_{4ij}\beta_i\beta_j], \quad (2)$$

where ε represents the error, Y represents the actual output, α_0 represents a constant, α_{1i}, α_{2i}, α_{3i}, and α_{4i} are regression coefficients, and β_i is a factor. The parameter combination improves when ε becomes closer to zero. The optimum parameter combination is identified and verified. If the experimental results are not as expected, the experiment must be redesigned.

2.2 Convolutional TFNN

The proposed CTFNN model, which is a combination of a CNN and TFNN, is used as a classifier. The CTFNN contains four layers: an input layer, a three-layer convolution layer, a feature fusion layer, and a TFNN layer. The TFNN contains four types of node, namely, fuzzification, rule, consequent, and output nodes. The architecture of the CTFNN model is described in the following paragraphs and illustrated in Fig. 2.

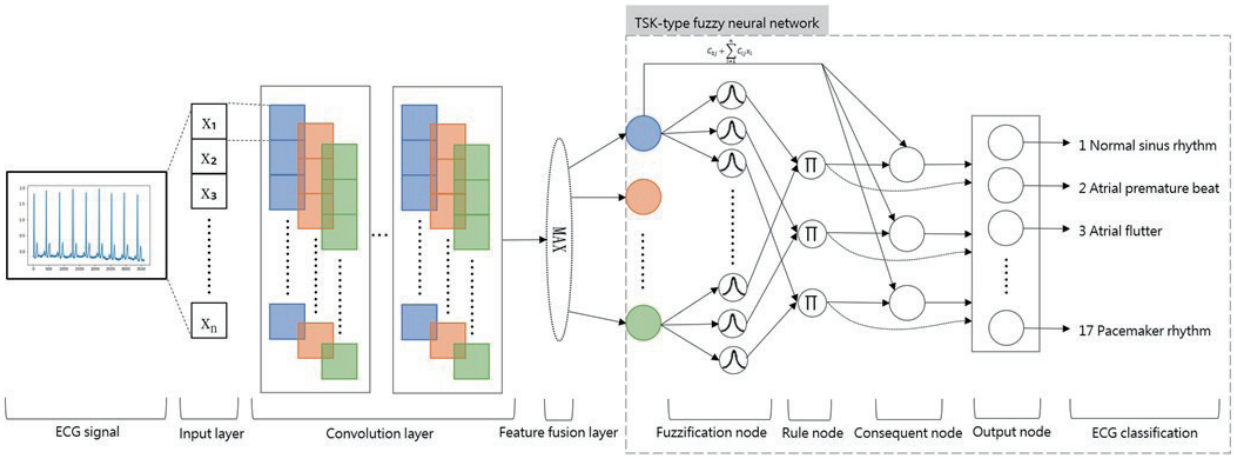


Fig. 2. (Color online) Proposed CTFNN architecture.

(1) Input layer

The network input is added to the input layer. In this study, one-dimensional ECG signals with a duration of 10 s and 3600 sampling points were used as input (Fig. 3; denoted as x_i).

(2) Convolutional layer

The convolutional layer slides the input signal through a convolution kernel. Three convolutional layer operations are performed to extract features, and the convolution operation is expressed as

$$y[f] = \sum_{k=1}^{ks-1} x[i+k+s] * w[f,k], \tag{3}$$

where $y[f]$ represents the output feature map and quantity, k is the index of the convolution kernel, ks is the size of the convolution kernel, and $w[f, k]$ is the convolution kernel.

(3) Feature fusion layer

Features are fused in the feature fusion layer. In the global max pooling operation, the convolutional layer's output is used to perform feature fusion and reduce the dimensionality of the feature map. The global max pooling operation is defined as

$$y_{GMP}[f] = \mathrm{Max}\left(y[f]\right), \tag{4}$$

where $y[f]$ represents the size of feature maps in the convolutional layer's output.

(4) TFNN

A TFNN, which is widely used in many fields, requires only a few fuzzy rules to solve complex nonlinear problems.[19–21] The TFNN architecture used in this study uses four types of node: fuzzification, rule, consequent, and output nodes. The fuzzy rules of the TFNN are expressed as

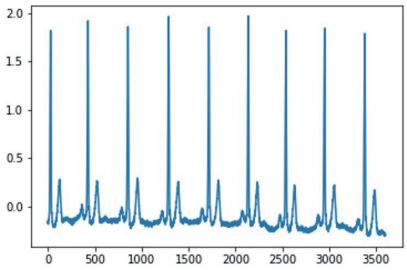


Fig. 3. (Color online) Example of an ECG signal.

$$R_j : \text{IF } x_1 \text{ is } A_{1j} \text{ and } x_2 \text{ is } A_{2j} \text{ and} \cdots \text{and } x_i \text{ is } A_{ij} \cdots \text{and } x_n \text{ is } A_{nj}$$
$$\text{THEN } y_j(x) = c_{0j} + c_{1j}x_1 + c_{2j}x_2 + \cdots + c_{ij}x_i + \cdots c_{nj}x_n,$$

where R_j is the jth fuzzy rule, x_1, x_2, ..., x_n is the input, A_{ij} is the fuzzy set, and c_{ij} is the parameter of the consequent.

- Fuzzification node

The fuzzy membership function is used in a fuzzy operation. The input signal is converted to a value between 0 and 1. In this study, a Gaussian membership function was used. This function is expressed as

$$o_{ij}^{(1)} = \mu_{ij}(x_i) = \exp\left(-\frac{1}{2}\left(\frac{x_i - m_{ij}}{\sigma_{ij}}\right)^2\right), \tag{5}$$

where m_{ij} is the mean and σ_{ij} is the deviation.

- Rule node

If the degree of membership of the input x_i is $\mu_{ij}(x_i)$, the firing strength of each rule is expressed as

$$o_j^{(2)} = \mu_j(x) = \prod_{j=1}^{r} o_{ij}^{(1)} \ , \ j = 1, 2, ..., r \ , \tag{6}$$

where r is the number of fuzzy rules.

- Consequent node

In this study, we adopted a TSK-type fuzzy rule, which is expressed as

$$o_j^{(3)} = o_j^{(2)}\left(c_{0j} + \sum_{i=1}^{n} c_{ij}x_i\right), \tag{7}$$

where n is the input dimension.

- Output node

This node completes the defuzzification operation and converts a fuzzy value into a crisp value. This operation is expressed as

$$o^{(4)} = \frac{\sum_{j=1}^{r} o_j^{(3)}}{\sum_{j=1}^{r} o_j^{(2)}} = \frac{\sum_{j=1}^{r} \mu_j(x)\left(c_{0j} + \sum_{i=1}^{n} c_{ij} x_i\right)}{\sum_{j=1}^{r} \mu_j(x)}. \quad (8)$$

3. Experimental Results

To evaluate the effectiveness of the proposed CTFNN model, experiments were performed using the MIT-BIH arrhythmia database.[22] First, the MIT-BIH arrhythmia database is introduced. Subsequently, the use of UED to optimize the CTFNN parameters as well as the experimental results and related experimental parameter configurations are described. Finally, the optimized architecture of the CTFNN is discussed.

3.1 Data set

The data set used in this experiment is the MIT-BIH arrhythmia database, which contains data from 47 patients, including 48 half-hour, dual-channel ECG signals from a heart rate sensor. Abnormal heart rhythm signals account for approximately 30% of all the signals, which are divided into 17 categories, including normal heart rhythm and abnormal signals. For measuring cardiac activity, only modified lead II records were used in this study. Because record numbers 102 and 104 did not use modified lead II records, data from only 45 patient records were used. Each ECG signal is 10 s long and contains 3600 sampling points for signal segmentation. A total of 1000 records were used in the experiment. The ratio of training data to validation data to test data was 70:15:15. The details of each signal category are presented in Table 1.

Table 1
Information on signal categories of MIT-BIH arrhythmia database.

No.	Classification	Symbol	Train	Val	Test	Total
1	Normal sinus rhythm	NSR	200	47	36	283
2	Atrial premature beat	APB	44	10	12	66
3	Atrial flutter	AFL	13	3	4	20
4	Atrial fibrillation	AFIB	96	21	18	135
5	Supraventricular tachyarrhythmia	SVTA	9	2	2	13
6	Pre-excitation	WPW	15	4	2	21
7	Premature ventricular contraction	PVC	98	19	16	133
8	Ventricular bigeminy	Bigeminy	38	8	9	55
9	Ventricular trigeminy	Trigeminy	10	2	1	13
10	Ventricular tachycardia	VT	7	1	2	10
11	Idioventricular rhythm	IVR	7	2	1	10
12	Ventricular flutter	VFL	6	1	3	10
13	Fusion of ventricular and normal beat	Fusion	7	3	1	11
14	Left bundle branch block beat	LBBBB	73	11	19	103
15	Right bundle branch block beat	RBBBB	45	8	9	62
16	Second-degree heart block	SDHB	6	3	1	10
17	Pacemaker rhythm	PR	26	4	14	45
	Total		700	150	150	1000

3.2 Experimental results obtained using CTFNN with UED

UED was used to select the optimal parameter combination. A total of 10 affecting factors were selected, including kernel size, the number of filters, the stride of the three convolutional layers, and the number of rule nodes in the TFNN. Three 4-level factors, four 3-level factors, and three 2-level factors were selected. To achieve effective uniformity, the number of experiments must be greater than twice the number of factors. Therefore, we conducted 21 experiments. The factors and levels of UED are presented in Table 2.

Next, according to the level number of the selected factor and the number of experiments, the corresponding initial UED table was designed and is shown in Table 3.

According to the initial UED table, the good lattice point method was used to establish a UED table and conduct experiments with this table. To improve reliability, each parameter combination was tested three times, and the average accuracy was calculated. The experimental results are presented in Table 4.

Multiple regression analysis was performed on the experimental results to obtain the optimal experimental parameters. Table 5 shows the optimal parameters of the proposed CTFNN model as selected using the UED method. The optimal experimental parameters for convolutional layer 1 are eight filters, a kernel size of 120, and a stride of 1; those for convolutional layer 2 are 64 filters, a kernel size of 30, and a stride of 1; those for convolutional layer 3 are 256 filters, a kernel size of 30, and a stride of 1; and the optimal parameter for the TFNN is 64 rule nodes.

3.3 Model analysis and discussion

To evaluate the performance of the CTFNN model, four evaluation indicators were used, namely, accuracy, recall, precision, and F1-score. The formulas for these indicators are

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}, \tag{9}$$

Table 2
Factors and levels of UED.

No.	Affecting factors		Level 1	Level 2	Level 3	Level 4
A	Convolution layer 1	Filter	8	16	32	64
B		Kernel size	30	60	120	–
C		Stride	1	2	–	–
D	Convolution layer 2	Filter	16	32	64	128
E		Kernel size	15	30	60	–
F		Stride	1	2	–	–
G	Convolution layer 3	Filter	32	64	128	256
H		Kernel size	15	30	60	–
I		Stride	1	2	–	–
J	TFNN	Rule node	32	64	128	–

Table 3
Initial UED table.

No.	A	B	C	D	E	F	G	H	I	J
1	1	9	11	9	2	3	18	18	7	8
2	2	19	14	13	17	21	15	6	14	9
3	3	3	9	8	19	7	4	3	11	17
4	4	12	18	19	12	14	5	21	12	3
5	5	11	4	2	15	12	17	8	3	20
6	6	14	5	21	10	2	10	5	21	12
7	7	1	7	11	6	16	16	9	19	2
8	8	20	20	12	7	4	6	10	5	19
9	9	2	17	4	9	20	11	17	2	13
10	10	18	3	3	5	9	2	15	15	5
11	11	7	21	17	20	10	19	16	18	14
12	12	8	1	15	14	18	7	2	6	6
13	13	16	16	5	1	13	12	1	17	15
14	14	17	10	16	18	5	13	13	1	1
15	15	15	8	20	4	19	20	12	10	18
16	16	6	15	1	16	1	8	11	16	7
17	17	21	6	7	21	15	9	19	8	11
18	18	4	13	18	3	11	1	7	4	10
19	19	5	2	14	8	6	14	20	13	16
20	20	13	19	6	11	8	21	4	9	4
21	21	10	12	10	13	17	13	14	20	21

Table 4
UED table and experimental results.

No.	A	B	C	D	E	F	G	H	I	J	Y1	Y2	Y3	Yavg
1	8	120	1	64	30	2	64	30	1	32	0.9467	0.9800	0.9733	0.9667
2	16	30	2	128	60	1	128	30	1	64	0.9733	0.9599	0.9666	0.9666
3	32	120	1	32	15	2	256	60	1	32	0.9666	0.9733	0.9599	0.9666
4	64	120	2	32	60	1	32	60	1	32	0.9133	0.9200	0.9200	0.9178
5	8	60	2	32	60	2	32	15	2	128	0.9399	0.9200	0.9400	0.9333
6	16	60	1	16	60	2	64	60	1	64	0.9666	0.9467	0.9533	0.9555
7	32	30	1	128	15	2	256	15	2	64	0.9266	0.9266	0.9200	0.9244
8	64	60	2	128	30	1	64	15	1	64	0.9466	0.9399	0.9533	0.9466
9	8	60	1	128	30	2	128	15	1	128	0.9399	0.9266	0.9466	0.9377
10	16	120	1	64	15	1	64	60	2	64	0.9666	0.9666	0.9533	0.9622
11	32	30	1	128	30	1	128	60	1	128	0.9533	0.9466	0.9599	0.9533
12	64	60	1	32	30	2	128	30	1	128	0.9599	0.9599	0.9266	0.9488
13	8	30	2	16	15	1	256	15	1	64	0.9133	0.9266	0.9333	0.9244
14	16	60	2	16	15	2	32	15	2	32	0.9066	0.8600	0.9200	0.8955
15	32	120	2	16	15	1	256	60	2	32	0.9599	0.9733	0.9666	0.9666
16	64	120	1	16	30	1	256	30	2	128	0.9666	0.9733	0.9599	0.9666
17	8	120	2	32	60	1	32	30	2	32	0.9533	0.9466	0.9200	0.9400
18	16	30	1	64	60	2	32	30	2	128	0.9333	0.9066	0.8866	0.9088
19	32	60	2	64	30	2	64	15	2	64	0.9200	0.9399	0.9533	0.9377
20	64	30	1	16	60	1	32	60	2	32	0.8666	0.9266	0.9066	0.8999
21	8	30	2	64	15	1	32	30	1	32	0.9399	0.9133	0.9266	0.9266

Table 5
Optimal experimental parameters for CTFNN model.

No.	Layer	Input	Filter	Kernel size	Stride	Rule	Activation
1	Input layer	3600	–	–	–	–	–
2	Convolution layer 1	–	8	120	1	–	ReLU
3	Convolution layer 2	–	64	30	1	–	ReLU
4	Convolution layer 3	–	256	30	1	–	ReLU
5	Feature fusion layer	–	256	–	–	–	–
6	Rule node	–	–	–	–	64	–
7	Output node	–		–	–	17	softmax

$$Recall = \frac{TP}{TP + FP}, \tag{10}$$

$$Precision = \frac{TP}{TP + FN}, \tag{11}$$

$$F1 - score = \frac{2 \times Recall \times Precision}{Recall + Precision}, \tag{12}$$

where *TP* is true positive, *TN* is true negative, *FP* is false positive, and *FN* is false negative. Figure 4 illustrates the confusion matrix of the CTFNN model. The proposed method was applied in the classification of 17 types of heart arrhythmia signal. A total of 14 types of heart arrhythmia signal were classified with an accuracy rate of 100%. The accuracy rates for the classification of the SDHB, PVC, and NSR arrhythmia signals (the remaining three signal types) were 91.67, 88.89, and 81.25%, respectively.

The performance of the CTFNN model was evaluated by comparing it with those of several widely used deep learning networks, such as LeNet,[23] AlexNet,[24] GoogLeNet,[25] ResNet18,[26] and T-CNFN.[27] Li *et al.*[27] used T-CNFN for the detection of arrhythmia. T-CNFN combines the Taguchi method and a convolutional neuro-fuzzy network. In T-CNFN, they adopted Mamdani-type fuzzy inference systems. The corresponding experimental results are presented in Table 6. The accuracy, recall, precision, and F1-score of the CTFNN were 97.33, 97.96, 96.00, and 96.97%, respectively, and the CTFNN achieved higher performance than did the other deep learning networks. In addition, because the CTFNN uses a TFNN instead of a fully connected neural network in the classifier, the number of parameters for the CTFNN was only 558728 (Table 6), which was less than those for LeNet, AlexNet, GoogLeNet, and ResNet18. CTFNN had more parameters than did T-CNFN but achieved higher accuracy, recall, precision, and F1-score than did T-CNFN.

In this study, we also compared the CTFNN with other conventional machine learning and deep learning methods by using the MIT-BIH arrhythmia database, and the corresponding experimental results are presented in Table 7. Oh *et al.*[30] proposed a CNN combined with a long short-term memory network that achieved an accuracy of 98.1%; however, they only categorized

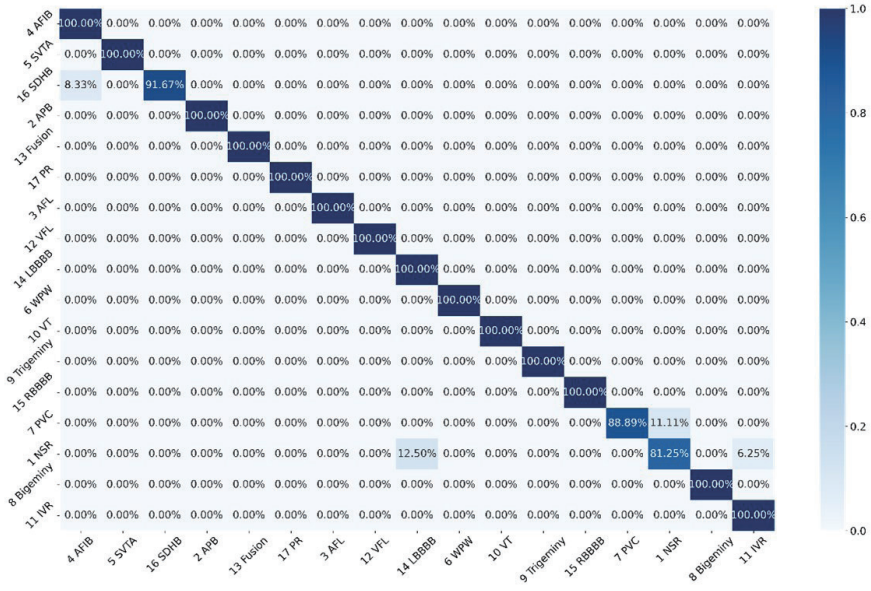


Fig. 4. (Color online) Confusion matrix of CTFNN model.

Table 6
Performance comparison of several deep learning networks.

Method	Accuracy (%)	Recall (%)	Precision (%)	*F*1-score (%)	Total parameters
LeNet[23]	30.67	32.17	30.67	31.40	1734501
AlexNet[24]	63.33	64.83	62.67	63.73	16233489
GoogLeNet[25]	77.33	80.88	73.33	76.92	21650217
ResNet18[26]	91.33	91.89	90.67	91.28	3867025
T-CNFN[27]	93.95	96.03	93.47	94.30	274657
Proposed method	97.33	97.96	96.00	96.97	558728

Table 7
Performance comparison of several conventional machine learning and deep learning methods.

Method	Length of signal	No. of classes	Classifier	Accuracy (%)
Pławiak[31]	3600 samples (10 s)	13	Evolutionary-neural system (based on support vector machine)	94.60
		15		91.28
		17		90.20
Acharya *et al.*[29]	360 samples (1 s)	5	CNN	94.03
Zubair *et al.*[32]	360 samples (1 s)	5	CNN	92.7
Mahmud *et al.*[33]	360 samples (1 s)	5	CNN and Inception	97.3
Oh *et al.*[30]	Variable length	5	CNN and LSTM	98.10
Yildirim *et al.*[28]	3600 samples (10 s)	13	CNN	95.20
		15		92.51
		17		91.33
Li *et al.*[27]	3600 samples (10 s)	17	T-CNFN	93.95
Proposed method	3600 samples (10 s)	17	CTFNN	97.33

five arrhythmias. Among the four models in Table 7 adopted for classifying 17 arrhythmia categories by using long-term (10 s) signal samples,[27,28,31] the CTFNN had the highest accuracy (97.33%). The other three models had accuracies of 90.20, 91.33, and 93.95%.[27,28,31]

4. Conclusions

In this paper, a CTFNN model is proposed for solving the problems faced in arrhythmia signal classification. The CTFNN is a combination of a CNN and TFNN. A TFNN is used instead of a fully connected neural network, which reduces the number of parameters and enables the model to mimic the human brain when classifying signals. The UED method is used to find the optimal parameter combination through a small number of experiments. The CTFNN was applied in the classification of 17 types of heart arrhythmia signal originating from heart rate sensors. The accuracy, recall, precision, and F1-score of the proposed CTFNN model for long-term signals were 97.33, 97.96, 96.00, and 96.97%, respectively, according to our experiments. The CTFNN outperformed several other methods. In addition, the number of parameters for the CTFNN was 558728, which was less than that for the LeNet model (i.e., 1,734,501). This network also classified 14 types of heart arrhythmia signal with an accuracy of 100%.

The accuracy rates of the CTFNN for the classification of the SDHB, PVC, and NSR arrhythmia signals (the remaining three types of signal) were 91.67, 88.89, and 81.25%, respectively. The number of arrhythmia signals in these three categories is small; therefore, obtaining their characteristics during learning is difficult. Consequently, in future research, we will introduce a generative adversarial network into the ECG signal classification model.[34] Unbalanced arrhythmia signals can be used for data augmentation to improve the classification ability of the model. In addition, future studies can combine the CTFNN with a hardware implementation for the real-time detection and display of heart rhythm.

References

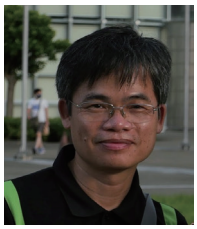
1 C. U. Kumari, A. S. D. Murthy, B. L. Prasanna, M. P. P. Reddy, and A.K. Panigrahy: Mater. Today: Proc. **45** (2021) 1393. https://doi.org/10.1007/s00521-022-06889-z
2 M. Li, X. J. Ma, C. Chen, Y. S. Yuan, S. L. Zhang, Z. W. Yan, C. Chen, F. F. Chen, Y. J. Bai, P. Y. Zhou, X. Y. Lv, and M. G. Ma: IEEE Access **9** (2021) 53687. https://ieeexplore.ieee.org/document/9395071
3 C. Y. Zou, A. Müller, U. Wolfgang, D. Rückert, P. Müller, M. Becker, A. Steger, and E. Martens: IEEE J. Transl. Eng. Health Med. **10** (2022) 1. https://ieeexplore.ieee.org/document/9869872
4 H. Yang and Z.Q. Wei: IEEE Access **8** (2020) 47103. https://ieeexplore.ieee.org/document/9027930
5 Z. Wu, T. Lan, C. Yang, and Z. Nie: IEEE Access **7** (2019) 170820. https://ieeexplore.ieee.org/abstract/document/8913462
6 T. Mahmud, S. A. Fattah, and M. Saquib: IEEE Access **8** (2020) 104788. https://ieeexplore.ieee.org/document/9104710.
7 Z. Zhang, Z. Zhang, C. Zou, Z. Pei, Z. Yang, J. Wu, S. Sun, and F. Gu: IEEE. Trans. Biomed. Eng. **70** (2023) 446. https://ieeexplore.ieee.org/document/9840907
8 L. A. Zadeh: Inf. Control. **8** (1965) 338. https://doi.org/10.1016/S0019-9958(65)90241-X
9 J. S. R. Jang: IEEE Trans. Syst. Man Cybern. (Syst.) **23** (1993) 665. https://ieeexplore.ieee.org/document/256541
10 C. Y. Wu, Y. W. Tsay, and A. C. C. Shih: Proc. 2022 Int. Conf. Advanced Robotics and Intelligent Systems (ARIS) (IEEE, 2022) 1–6. https://ieeexplore.ieee.org/document/9910444

11 Z. Boreiri, A. N. Azad, and A. Ghodousian: 2022 27th Int. Conf. Computer Conf. Computer Society of Iran (CSICC) (2022) 1. https://ieeexplore.ieee.org/document/9780525
12 G. Du, Z. Wang, C. Li, and P. X. Liu: IEEE Trans. Fuzzy Syst. **29** (2021) 2100. https://ieeexplore.ieee.org/document/9088236
13 G. Taguchi: IEEE Trans. Reliab. **44** (1995) 225. https://ieeexplore.ieee.org/document/387375
14 G. Taguchi, S. Chowdhury, and Y. Wu: Taguchi's Quality Engineering Handbook (New York, USA Wiley, 2005).
15 W. C. Weng: IEEE Access **10** (2022) 107242. https://ieeexplore.ieee.org/document/9913934
16 K. T. Fang and D. K. J. Lin: Handb. Stat. **22** (2003) 131. https://doi.org/10.1016/S0169-7161(03)22006-X
17 J. T. Tsai, P. Y. Yang, and J. H. Chou: IEEE Access **6** (2018) 40365. https://ieeexplore.ieee.org/document/8412479
18 S. F. Yang and W. T. K. Chien: IEEE Trans. Reliab. **64** (2015) 1158. https://ieeexplore.ieee.org/document/7160791
19 J. Guan, C. M. Lin, G. L. Ji, L. W. Qian, and Y. M. Zheng: IEEE Access **6** (2018) 1670. https://ieeexplore.ieee.org/document/8141861
20 C. S. Ouyang, W. J. Lee, and S. J. Lee: IEEE Trans. Syst. Man Cybern. **35** (2005) 751. https://ieeexplore.ieee.org/document/1468248
21 S. Nemet, D. Kukolj, G. Ostojić, S. Stankovski, and D. Jovanović: Appl. Intell. **49** (2019) 3909. https://doi.org/10.1007/s10489-019-01485-6
22 G. B. Moody and R. G. Mark: IEEE Eng. Med. Biol. Mag. **20** (2001) 45. https://ieeexplore.ieee.org/document/932724
23 Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner: Proc. IEEE **86** (1998) 2278. https://ieeexplore.ieee.org/document/726791
24 A. Krizhevsky, I. Sutskever, and G. E. Hinton: Proc. 25th Int. Conf. Neural Information Processing Systems (NIPS, 2012) 1097–1105. https://doi.org/10.1145/3065386
25 C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich: Proc. 2015 IEEE Conf. Computer Vision and Pattern Recognition (CVPR) (IEEE, 2015) 1–9. https://ieeexplore.ieee.org/document/7298594
26 K. He, X. Zhang, S. Ren, and J. Sun: 2016 IEEE Conf. Computer Vision and Pattern Recognition (CVPR) (IEEE, 2016) 770. https://doi.org/10.48550/arXiv.1512.0338
27 J. Li, J. Y. Jhang, C. J. Lin, and X. Q. Lin: Sens. Mater. **34** (2022) 2853. https://doi.org/10.18494/SAM3924
28 Ö. Yildirim, P. Plawiak, R. S. Tan, and U. R. Acharya: Comput. Biol. Med. **102** (2018) 411. https://doi.org/10.1016/j.compbiomed.2018.09.009
29 U. R. Acharya, S. L. Oh, Y. Hagiwara, J. H. Tan, M. Adam, A. Gertych, and R. S. Tan: Comput. Biol. Med. **89** (2017) 389. https://doi.org/10.1016/j.compbiomed.2017.08.022
30 S. L. Oh, E. Y. K. Ng, R. S. Tan, and U. R. Acharya: Comput. Biol. Med. **102** (2018) 278. https://doi.org/10.1016/j.compbiomed.2018.06.002
31 P. Plawiak: Expert Syst. Appl. **92** (2018) 334. https://doi.org/10.1016/j.eswa.2017.09.022
32 M. Zubair, J. Kim, and C. Yoon: Proc. 2016 6th Int. Conf. IT Convergence and Security (ICITCS) (IEEE, 2016) 1–5. https://ieeexplore.ieee.org/document/7740310
33 T. Mahmud, A. R. Hossain, and S. A. Fattah: Proc. 2019 7th Int. Conf. Robot Intelligence Technology and Applications (RiTA) (IEEE, 2019) 32. https://ieeexplore.ieee.org/document/8932850
34 T. H. Rafi and Y. W. Ko: IEEE Access **10** (2022) 100501. https://ieeexplore.ieee.org/document/9889702

About the Authors

Cheng-Jian Lin received his B.S. degree in electrical engineering from Ta Tung Institute of Technology, Taipei, Taiwan, R.O.C., in 1986 and his M.S. and Ph.D. degrees in electrical and control engineering from National Chiao Tung University, Taiwan, R.O.C., in 1991 and 1996, respectively. Currently, he is a chair professor of the Computer Science and Information Engineering Department, National Chin-Yi University of Technology, Taichung, Taiwan, R.O.C. His current research interests are in machine learning, pattern recognition, intelligent control, image processing, intelligent manufacturing, and evolutionary robots. (cjlin@ncut.edu.tw)

Han Cheng received her B.S. degree from the Department of Business Management, National Taichung University of Science and Technology, Taichung, Taiwan, R.O.C., in 2022. Currently, she is a graduate student of the Department of Computer Science and Information Engineering, National Chin-Yi University of Technology, Taichung, Taiwan. Her current research interests include deep learning, neural fuzzy systems, and computer vision and applications. (4b117051@gmail.com)



Chun-Lung Chang received his M.S. degree in power mechanical engineering from National Tsing-Hua University, Hsinchu, Taiwan, R.O.C., in 1992 and his Ph.D. degree in electrical and control engineering from National Chiao-Tung University, Hsinchu, Taiwan, R.O.C., in 2006. He is an associate professor in National Chin-Yi University of Technology, Taichung, Taiwan, R.O.C. His current research interests are in artificial intelligence, image processing, computer vision, and smart manufacturing.
(VincentChang0304@ncut.edu.tw)