JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 34, 1425-1440 (2018)
DOI: 10.6688/JISE.201811_34(6).0005

Linear-Discriminant-Analysis-Based Type-2 Fuzzy Neural-Network for Speech Detection and Recognition

GIN-DER WU AND ZHEN-WEI ZHU
Department of Electrical Engineering
National Chi Nan University
Nantou, 545 Taiwan
E-mail: ginderwu@ncnu.edu.tw

Two important classification problems in speech signal processing are speech detection and recognition. They are easily affected by noisy environments where there may exist concurrent noises due to movements of desks, door slams, *etc.* To solve this problem, a linear-discriminant-analysis-based type-2 fuzzy-neural-network (LDA2FNN) is proposed. In handing noisy data with uncertainties, type-2 fuzzy-systems generally outperform their type-1 counterparts. Therefore, type-2 fuzzy-sets are used in the antecedent parts to cope with the noisy data. The most important consideration for classification problems is the "discriminability". To increase the "discriminability", linear-discriminant-analysis (LDA) is applied in the consequent parts. Compared with other existing fuzzy neural networks, the novelty of the proposed LDA2FNN is its consideration of both uncertainty and discriminability. Furthermore, its computation load is low. In experiments, LDA2FNN is successfully applied to speech detection and recognition. Experimental results indicate that the proposed LDA2FNN performs better than the other fuzzy neural networks.

Keywords: classification, speech detection and recognition, linear-discriminant-analysis, type-2 fuzzy-neural-network, uncertainty

1. INTRODUCTION

In noisy speech classification problems, fuzzy-rule-based methods have received considerable attention. Although these fuzzy-based classifiers try to minimize the training error, noise usually increases the uncertainty and reduces the discriminability. Therefore, discriminability and uncertainty are very crucial to noisy data classification. As far as discriminability is concerned, principal component analysis (PCA) [1, 2] has been used in the optimization of classification. Some studies propose a self-constructing neural fuzzy inference network (SONFIN) using PCA to classification problems [3, 4]. However, PCA lacks the analysis of the statistics among different classes, explaining why the discriminative capability of PCA is weak.

Linear discriminant analysis (LDA) is a powerful method to optimize the discriminative capability among different classes. Because the discriminative capability critically determines the performance of classification, LDA has been adopted to classify highly confusable patterns. By using LDA, the study of [5] proposed a maximizing-discriminability-based self-organizing fuzzy network (MDSOFN) which can classify highly confusable patterns. Experimental results indicate that the LDA-derived fuzzy network is better than the PCA-based fuzzy network and support vector machine (SVM) based

Received September 21, 2017; revised November 24, 2017; accepted February 2, 2018.
Communicated by Chung-Hsien Wu.

fuzzy network [6]. Other studies [7, 8] adopted minimum classification error (MCE) methods, where MCE is used to increase the temporal discriminative capability rather than to fit the distributions to the data. This study adopts LDA in the consequent parts of the proposed type-2 fuzzy neural network (LDA2FNN) to increase the discriminability for solving classification problems.

To optimize fuzzy neural networks, some state-of-the-art hybrid machine learning systems use either particle-swarm optimization (PSO) [9-11] or ant-colony optimization (ACO) [12, 13]. Type-2 fuzzy systems can model and minimize the effects of uncertainties in rule-based systems [14-19]. Type-2 fuzzy logic systems (FLS) outperform their type-1 counterparts in handing problems with uncertainties such as noisy data. This ability is attributed to type-2 fuzzy-sets, which have 3-D membership functions. The third dimension in type-2 fuzzy-sets and a footprint of uncertainty provide an additional degree of freedom for type-2 FLS to model and handle uncertainties. Therefore, the antecedent parts of the proposed LDA2FNN uses type-2 fuzzy-sets to model the uncertainty.

Deep learning is a popular method in artificial intelligence. Deep neural network (DNN) has shown good performance in speech signal processing [20-23]. Each network layer is regarded as a different feature space of the input data. The DNN uses back propagation algorithm with feed-forward multi-layer neural networks. Compared with FNN, DNN needs more powerful CPU to process the data in networks. Additionally, unlike fuzzy rules, a DNN does not have much "interpretable" information. Hence, this study focuses on type-2 FNN.

The rest of this paper is organized as follows. Section 2 introduces the optimization of linear-discriminant-analysis (LDA) type-2 fuzzy rule for noisy speech classification. Section 3 discusses the structure and parameters of the proposed LDA2FNN. Section 4 verifies the performance of LDA2FNN by using speech detection and recognition. Finally, Section 5 makes the conclusions.

2. FUZZY FOR NOISY SPEECH CLASSFICATION

This section discusses the optimization of fuzzy rules for noisy data classification. The proposed linear-discriminant-analysis (LDA) type-2 fuzzy rule includes an LDA-matrix which is introduced as follows.

2.1 Linear-Discriminant-Analysis Type-2 Fuzzy

The type-2 fuzzy rule with linear-discriminant-analysis is shown as follows.

Rule r: IF x_1 is $\tilde{A}_1^r$ AND…AND x_N is $\tilde{A}_N^r$

$$\text{THEN } y \text{ is } \tilde{a}_0^r + \sum_{n=1}^{p} \tilde{a}_n^r t_n, \quad r = 1, ..., M \tag{1}$$

where $x_1 \dots x_N$ represent input variables; $\tilde{A}_1^r \dots \tilde{A}_N^r$ are interval type-2 fuzzy sets; y is the output of fuzzy rule r; M is the number of rules, and $\tilde{a}_n^r = [c_n^r - s_n^r, c_n^r + s_n^r]$, $n = 0, \dots, p$ are interval sets. Consequent parts $t_n(n = 1, \dots, p)$ are updated by the LDA-matrix, $W_{LDA} \in \mathbb{R}^{N \times p}$ as follows.

$$[t_1\ t_2\ \ldots\ t_p]^T = W_{LDA}{}^T \cdot [x_1\ x_2\ \ldots\ x_N]^T \tag{2}$$

To derive the LDA-matrix, the mean vector $\mu^{(j)}$ and covariance matrix $\sum^{(j)}$ labeled as belonging to class j are then computed. The total number of classes is J, and the input vector $X^{(j)}(n)$ in the training set is labeled as the jth class. N_j indicates the total number of such input vectors $X^{(j)}(n)$ labeled as belonging to the jth class.

$$\mu^{(j)} = \frac{1}{N_j}\sum_{n=1}^{N_j} X^{(j)}(n) \tag{3}$$

$$\sum\nolimits^{(j)} = \frac{1}{N_j}\sum_{n=1}^{N_j}(X^{(j)}(n)-\mu^{(j)})(X^{(j)}(n)-\mu^{(j)})^T \tag{4}$$

2.2 LDA-Matrix

LDA transformation can be regarded as a change of input coordinates. By using the concept of between-class and within-class in Fig. 1, LDA can seek directions that are efficient for discrimination. $W_{LDA} \in \mathbb{R}^{N\times p}$ is determined by the between-class matrix S_B and within-class matrix S_W which are defined as follows.

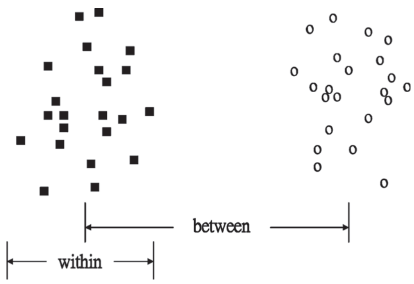


Fig. 1. Between-class and within-class concept.

$$S_B = \sum_{j=1}^{J} N_j(\mu^{(j)}-\mu)(\mu^{(j)}-\mu)^T \tag{5}$$

$$\mu = \left(1\Big/\sum\nolimits_{j=1}^{J} N_j\right)\sum\nolimits_{j=1}^{J} N_j\mu^{(j)} \tag{6}$$

$$S_W = \sum_{j=1}^{J} N_j \sum\nolimits^{(j)} \tag{7}$$

The optimal direction $e = \arg\max_e \dfrac{e^T S_B e}{e^T S_W e}$ is applied to increase discriminative capabilities among different classes.

$$S_W^{-1} S_B e = \lambda e \tag{8}$$

In this form, e denotes the eigenvector of matrix $S_W^{-1}S_B$. The discrimination matrix $W_{LDA} \in \mathbb{R}^{N\times p}$ gathers the eigenvectors of $S_W^{-1}S_B$ corresponding to the largest p eigenvalues.

3. LDA-BASED TYPE-2 FUZZY NEURAL NETWORK

This section describes the structure of LDA2FNN, which attempts to find the most discriminative representation of a fuzzy neural network in speech detection and classification. The proposed LDA2FNN is a five-layered network. Fig. 2 illustrates the case of a single output of LDA2FNN.

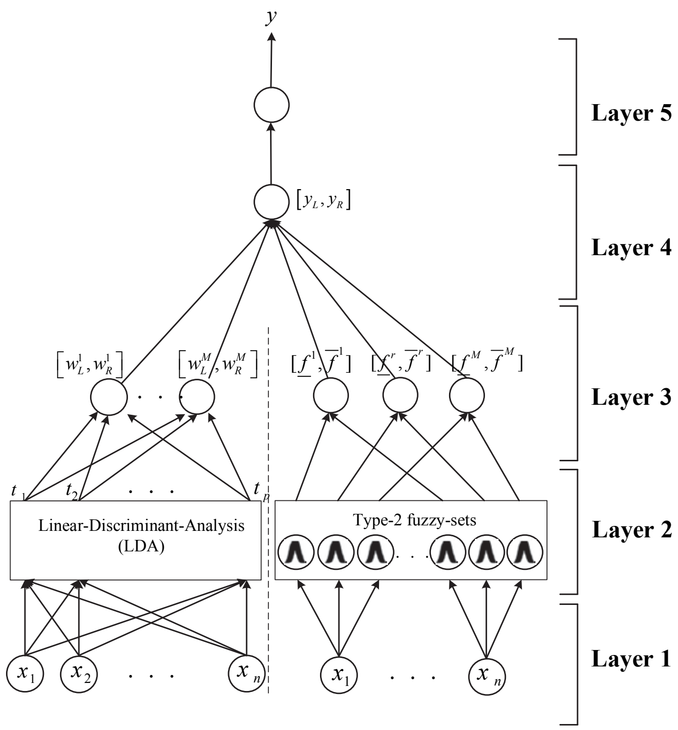


Fig. 2. Linear-discriminant-analysis-based type-2 fuzzy neural network (LDA2FNN).

3.1 Structure of LDA2FNN

Layer 1: The inputs nodes are range normalization. Each node in this layer scales input x_n, $n = 1, \ldots, N$ to within the range [−1, 1]. There are no weights to be adjusted in this layer.

Layer 2: An interval type-2 Membership Function (MF) for the rth fuzzy set $\tilde{A}_n^r$ in input variable x_n has a standard deviation σ and an uncertain mean $[m_1, m_2]$ as follows.

$$\tilde{\mu}_n^r = \exp\left[-\left(\frac{x_n - \tilde{m}_n^r}{\sigma_n^r}\right)^2\right] = N\left(\tilde{m}_n^r, \sigma_n^r; x_n\right), \tilde{m}_n^r \in \left[m_{n1}^r, m_{n2}^r\right] \tag{9}$$

The upper MF, $\bar{\mu}_n^r$, and lower MF, $\underline{\mu}_n^r$ are

$$\bar{\mu}_n^r = \begin{cases} N\left(m_{n1}^r, \sigma_n^r; x_n\right), & x_n < m_{n1}^r \\ 1, & m_{n1}^r \le x_n \le m_{n2}^r \\ N\left(m_{n2}^r, \sigma_n^r; x_n\right), & x_n > m_{n2}^r \end{cases}, \tag{10}$$

$$\underline{\mu}_n^r = \begin{cases} N\left(m_{n2}^r, \sigma_n^r; x_n\right), & x_n \le \dfrac{m_{n1}^r + m_{n2}^r}{2} \\ N\left(m_{n1}^r, \sigma_n^r; x_n\right), & x_n > \dfrac{m_{n1}^r + m_{n2}^r}{2} \end{cases}. \tag{11}$$

Based on the above equations, the output can be represented as the interval type $|\underline{\mu}_n^r, \ \bar{\mu}_n^r|$. In "LDA" part, t_n ($n = 1, \ldots, p$) are updated by $W_{LDA} \in \mathbb{R}^{N \times p}$.

Layer 3: This rule-node part runs the fuzzy meet operation by using an algebraic product operation. The output of a rule node represents its corresponding firing strength. This firing strength is calculated as $|\underline{f}^r, \ \bar{f}^r|$.

$$\bar{f}^r = \prod_{n=1}^{N} \bar{\mu}_n^r \tag{12}$$

$$\underline{f}^r = \prod_{n=1}^{N} \underline{\mu}_n^r \tag{13}$$

In consequent-node parts, the fuzzy set is represented by $[w_L^r, w_R^r]$ as follows.

$$\left[w_L^r, w_R^r\right] = \left[c_0^r - s_0^r, c_0^r + s_0^r\right] + \sum_{n=1}^{p} \left[c_n^r - s_n^r, c_n^r + s_n^r\right] t_n \tag{14}$$

That is

$$w_L^r = \sum_{n=0}^{p} c_n^r t_n - \sum_{n=0}^{p} |t_n| s_n^r, \tag{15}$$

$$w_R^r = \sum_{n=0}^{p} c_n^r t_n + \sum_{n=0}^{p} |t_n| s_n^r. \tag{16}$$

where $t_0 = 1$.

Layer 4: This layer implements the type reduction. The type-reduced set is an interval type-1 fuzzy set $[y_L, y_R]$. The outputs y_L and y_R can be calculated by using Kamik-Mendel iterative procedure [24].

Layer 5: Based on the above interval type-1 set $[y_L, y_R]$, this layer implements the defuzzification operations. The defuzzified output is shown as follows.

$$y = \frac{y_L + y_R}{2} \tag{17}$$

3.2 Structure Learning

A cluster in the input space corresponds to a rule. Therefore, a new rule is generated according to clustering on the input variables. The rule firing strength is calculated as follows.

$$\hat{f}^r = \frac{1}{2}\left(\bar{f}^r + \underline{f}^r\right) \tag{18}$$

Then, this firing strength center acts as a rule generation criterion. For each incoming data $X = (x_1, \ldots, x_N)$, find

$$I = \arg \max_{1 \le r \le M(t)} \hat{f}^r(X) \tag{19}$$

where $M(t)$ is the number of rules at time t. If $\hat{f}^I(X) \le \phi_{\text{th}}$, then a new rule is generated as $M(t+1) = M(t)+1$. $\phi_{\text{th}} \in (0,1)$ is a pre-specified threshold to decide the number of input clusters. Once a new cluster is generated, its initial uncertain mean in input variable x_n is set as

$$\tilde{m}_n^{M(t+1)} \in \left[m_{n1}^r, m_{n2}^r\right] = \left[x_n - 0.1, x_n + 0.1\right], n = 1, \ldots, N \tag{20}$$

A fuzzy set generation criterion is computed as

$$\hat{\mu}_n^r = \frac{1}{2}\left(\underline{\mu}_n^r + \bar{\mu}_n^r\right) \tag{21}$$

The nearest fuzzy set is defined as the one that possesses the maximum degree.

$$I_n = \arg \max_{1 \le r \le M(t)} \hat{\mu}_n^r, \quad n = 1, \ldots, N \tag{22}$$

The initial width of the first class is $\sigma_n^1 = \sigma_{init}$ which is set as a small value. The other initial widths are assigned as

$$\sigma_n^{M(t+1)} = \beta \left| x_n - \frac{1}{2}\left(m_{n1}^{I_n} + m_{n2}^{I_n}\right)\right| \tag{23}$$

where β determines the overlap degree between two fuzzy sets. Once a new rule is generated, generation of the corresponding consequent node follows. The initial consequent parameters are set to

$$[c_0^r - s_0^r, c_0^r + s_0^r] = [y^d - 0.1, y^d + 0.1], r = 1, \ldots, M \tag{24}$$

where $c_0^r = y^d$ is the desired output for input X. The initial parameter $s_0^r = 0.1$ determines the initial output interval range. The other initial consequent parameters c_n^r, $n = 1, \ldots, p + 1$, and initial s_n^r, $n = 1, \ldots, p$, are the same as s_0^r.

$$c_n^r = 0.01,\ s_n^r = s_0^r = 0.1,\ n = 1, \ldots, p, \text{ and } r = 1, \ldots, M \tag{25}$$

3.3 Parameter Learning

Considering the single-output case for clarity, our goal is to minimize the error function $E = ½[y - y^d]^2$. y and y^d denote real and desired outputs, respectively. Parameter learning of m_{n1}^r, m_{n2}^r and σ_n^r are discussed in [25].

$$m_{n1}^r(t+1) = m_{n1}^r(t) - \eta \frac{\partial E}{\partial m_{n1}^r} \tag{26}$$

$$m_{n2}^r(t+1) = m_{n2}^r(t) - \eta \frac{\partial E}{\partial m_{n2}^r} \tag{27}$$

$$\sigma_n^r(t+1) = \sigma_n^r(t) - \eta \frac{\partial E}{\partial \sigma_n^r} \tag{28}$$

In addition, the update rules for c_n^r and s_n^r are

$$c_n^r(t+1) = c_n^r(t) - \eta \frac{\partial E}{\partial c_n^r}, \tag{29}$$

$$s_n^r(t+1) = s_n^r(t) - \eta \frac{\partial E}{\partial s_n^r}. \tag{30}$$

To clarify the underlying reason of optimization equations, Fig. 3 shows a new global flowchart for optimization. Updating the first p terms by $W_{LDA} \in \mathbb{R}^{N \times p}$ improves the speed of convergence.

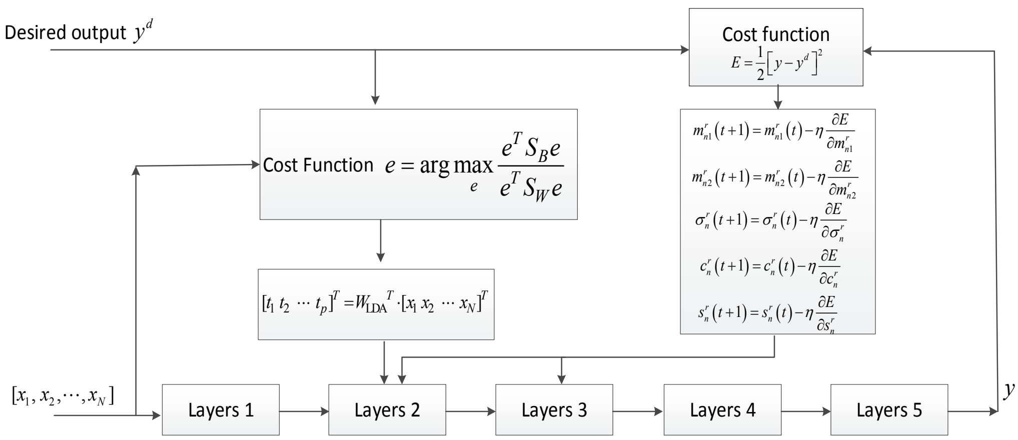


Fig. 3. The learning of parameters.

4. EXPERIMENTS

To test the performance of the proposed LDA2FNN, two noisy speech classification problems are applied in experiments. They are speech detection and recognition in noisy environments. A detailed comparative performance analysis is proposed for the fuzzy neural network with PCA [3], fuzzy neural network with LDA [5], and type-2 fuzzy neural network [25].

4.1 Noisy Speech Detection

The flowchart of speech detection using LDA2FNN is shown in Fig. 4. The sampling rate is 8KHz. The input features [4, 26] of the LDA2FNN consist of the average of the logarithmic root-mean-square (rms) energy on the first five frames of the recording interval (Noise_time), refined time-frequency (RTF) parameter, and zero-crossing rate (ZCR). These three features are normalized to be within [0, 1]. The output vector of LDA2FNN determines whether the corresponding frame is a speech signal or noise. The output vector use (1, 0) to stand for speech signal, and (0, 1) to stand for noise signal. The decoder in Fig. 4 decodes the output vector (1, 0) as a speech, and (0, 1) as noise. To eliminate the impulse noise, the output waveform of the decoder then passes through a median filter. Finally, the speech-waveform with sufficient magnitude and duration is defined as a speech-signal island.

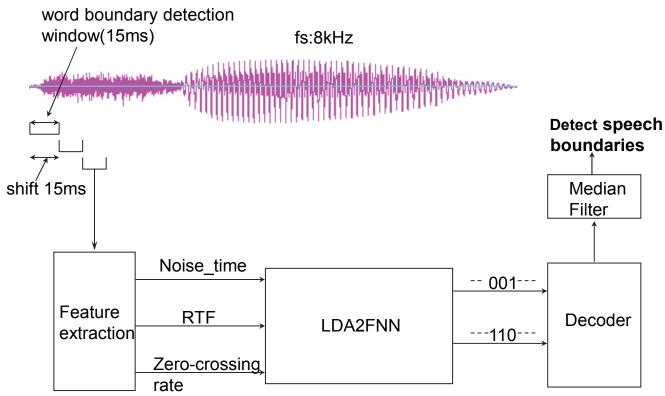


Fig. 4. The flowchart of speech detection using LDA2FNN.

In speech detection, isolated Mandarin digits (0~9) were used as training patterns. It is a frame-based detection with 15ms window (120-points). 60 training patterns (16-bit waveform) are selected from five SNR conditions (SNR = 0db, 5dB, 10dB, 15dB, 20dB). They are classified as speech or noises based on waveform, spectrum displays and audio output. In these 60 training patterns, 30 patterns are "speech" with a desired output vector of (1, 0), and the other 30 patterns are "noise" with a desired output vector of (0, 1). After training, LDA2FNN is ready for speech boundary detection.

One another continuous Mandarin sentence is used as testing pattern which is pronounced as "Jin Tian Tian Qi Hen Hao" in Fig. 5. The original speech waveform is shown in Fig. 5 (a) in which sample index 15000 belongs to the speech signal. The endpoint of speech signal locates about index 22600. They are classified as speech or noises based on waveform, spectrum displays and audio output. Fig. 5 (b) shows the speech waveform recorded in additive white noise (SNR = 0dB). Figs. 6 (a)-(c) show the speech endpoint locations detected by SONFIN-PCA, MDSOFN-LDA and SEIT2FNN, respectively. Those detection methods do not correctly classify the sample index 15000 as speech signal. Furthermore, their speech endpoints are far away from index 22600. Hence, LDA2FNN in Fig. 6 (d) is applied to solve the above errors. In LDA2FNN, type-2 fuzzy-sets are used in the antecedent parts to cope with the uncertainty. Besides, linear-discriminant-analysis (LDA) is applied in the consequent parts to increase the discriminability. Compared with other existing fuzzy neural networks, the novelty of the proposed LDA2FNN is its consideration of both uncertainty and discriminability. The endpoint detection of LDA2FNN in Fig. 6 (d) is more precise than those in Figs. 6 (a)-(c) because sample index 15000 is correctly classified as speech signal. Besides, its speech endpoint approaches 22600. To consider five noise types [27] (multi-talker babble noise, cockpit noise, noise on the floor of car factory, vehicle noise and white noise) with five SNR conditions (SNR = 0db, 5dB, 10dB, 15dB, 20dB), the detection performance of different fuzzy classifiers is shown in Table 1. The detection accuracy is frame-based unit. According to the experimental results, the proposed LDA2FNN performed better than the other methods.

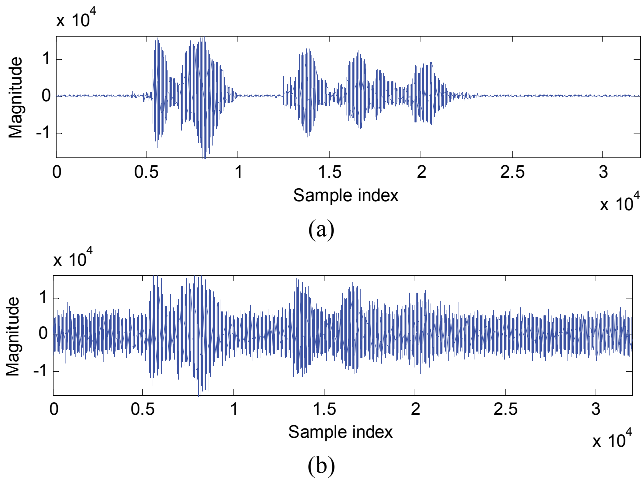
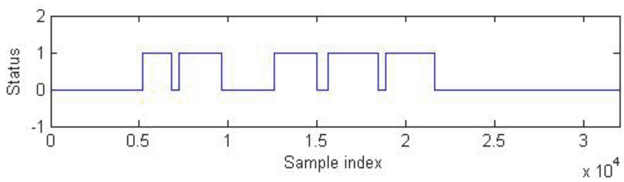
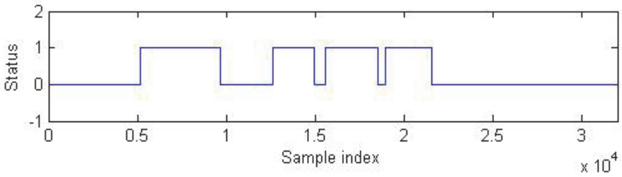


Fig. 5. (a) Original speech waveform; (b) Speech recorded in additive white noise (SNR=0dB).

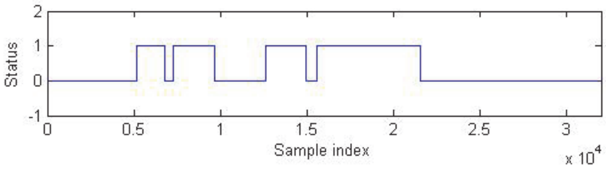


(a)

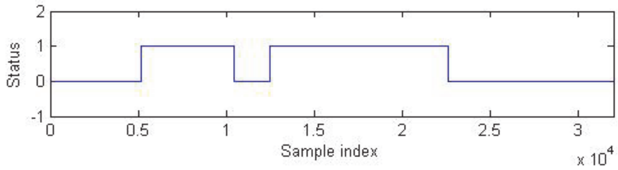
Fig. 6. (a) Speech detected by SONFIN-PCA.



(b)



(c)



(d)

Fig. 6. (b) Speech detected by MDSOFN-LDA; (c) Speech detected by SEIT2FNN; (d) Speech detected by LDA2FNN.

Table 1. Speech detection accuracy.

FNN Classifier	SONFIN-PCA [3]	MDSOFN-LDA[5]	SEIT2FNN [25]	LDA2FNN
Structure analysis	19 rules	17 rules	10 rules	9 rules
Number of parameters	275	173	250	156
Average correct rate	86.2%	88.6%	86.0%	89.8%

For further comparison, the performances of different methods of speech detection using the same training and test speech sequences are applied in Table 2. To compare with DNN, one more experiment is proposed. It is not easy to determine the number of hidden nodes in DNN. Since the decision rules learned by LDA2FNN is 9 for speech detection, the number of hidden nodes in DNN is also set to be 9. Based on the same input features, the parameters of DNN is updated by the gradient descent method. Its performance is worse than our proposed method. An adaptive long-term sub-band entropy for voice activity detection [28] is tested. Its thresholds are determined by trial and error. Without intelligent machine learning, its performance is worse than our proposed method. Then, the second and third methods adopted Haar wavelet energy and entropy features for speech detection [29]. They are HWEE-TRFN (Type-I recurrent fuzzy neural network) and HWEE-RSEIT-2FNN (Type-II recurrent fuzzy neural network). They both lack efficient analysis of the statistics among different classes, explaining why their performances are worse than our method.

Table 2. Speech detection accuracy.

Speech detection	ALT-SubEnpy[28]	HWEE-TRFN[29]	HWEE-RSEIT2FNN [29]	DNN	LDA2FNN
Average correct rate	85.8%	87.4%	88.2%	85.6%	89.8%

4.2 Noisy Speech Recognition

In the second experiment, the flowchart of speech detection using LDA2FNN is shown in Fig. 7. The input features of LDA2FNN adopt the modified two-dimensional cepstrum (MTDC) [30]. First, the speech signal goes through a first-order pre-emphasis filter with a pre-emphasis coefficient of 0.97. Then the speech signal multiplies by a Hamming window. By using discrete-fourier-transform (DFT) and mel-scale filter-bank, the energy of each frequency band is computed. To remove additive noise component along the

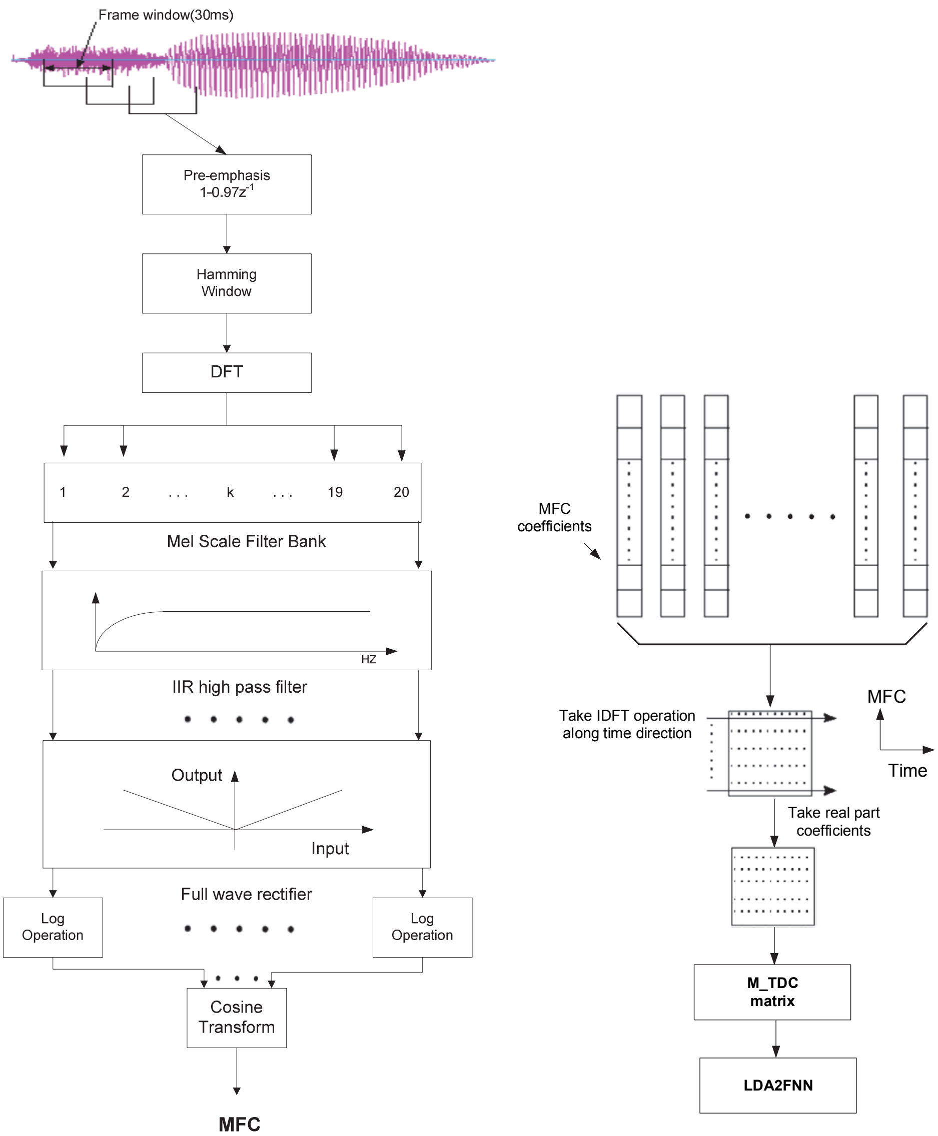


Fig. 7. The flowchart of speech recognition using LDA2FNN.

frame axis, the 4th-order infinite-impulse-response (IIR) temporal-filter and the half-wave rectification are used. Then MFC coefficients are obtained by taking the logarithm and cosine-transform. In this step, the MFC coefficients of each frame along the time axis are collected to form the MFC-time matrix. Finally, the MTDC matrix is generated from the real-part coefficients of the inverse discrete Fourier transform (IDFT) along the time axis. To represent a speech utterance, only 30 MTDC coefficients are selected to form a feature vector (x_i, i = 1, 2, …, 30). These 30 positions are randomly selected instead of using the genetic algorithm (GA).

In speech recognition, isolated Mandarin digits (0~9) were used as training and testing patterns. The 16-bit speech data are a set of isolated Mandarin digits (0~9) spoken by 10 speakers. The sampling rate is 8 kHz, and the frame size is 240 samples with 50% overlap. Training is performed by 2000 utterances. The other 2000 utterances are used for testing.

The noise is taken from the NOISEX-92. The attenuation is applied to ensure the addition of noise without causing an overflow of the 16-bit integer range. The recognition accuracy is word-based unit. To verify the performance, the recognition accuracy of different fuzzy classifiers averaged over five SNR conditions (SNR = 0db, 5dB, 10dB, 15dB, 20dB) and five noise types (multi-talker babble noise, cockpit noise, noise on the floor of car factory, vehicle noise and white noise) is shown in Table 3. Experimental results indicate that the proposed LDA2FNN performed better than the others.

For the purpose of comparison, we also studied other different speech recognition methods which use the same training and test speech sequences. Hidden–Markov-Model (HMM) is usually applied to speech recognition [8]. The state number is set to be 8. In HMM, the maximum likelihood (ML) is used to determine the optimal parameters. With the robust speech feature MFCC, the average recognition rate of HMM is about 76.6%. Furthermore, a Gaussian Mixture Model (GMM) [30] is applied for comparison. In the training phase, each model (Mandarin digit) is trained by a mixture of four Gaussian distribution density functions. The mixture number is set to be 4. To estimate the means and deviations of mixtures, the maximum likelihood (ML) estimation is applied. The goal is to maximize the likelihood function with respect to the parameters. According to Table 4, LDA2FNN is better than HMM and GMM because of the ability to consider both uncertainty and discriminability. To compare with DNN, one more experiment is proposed. It is not easy to determine the number of hidden nodes in DNN. Since the decision rules learned by LDA2FNN is 11 for speech detection, the number of hidden nodes in DNN is also set to be 11. Based on the same input features, the parameters of DNN is updated by the gradient descent method. Its performance is worse than our proposed method.

Table 3. Speech recognition accuracy.

FNN Classifier	SONFIN-PCA[3]	MDSOFN-LDA[5]	SEIT2FNN[25]	LDA2FNN
Structure analysis	24 rules	20 rules	12 rules	11rules
Number of parameters	2784	2550	8520	2210
Average correct rate	77.4%	80.2%	78.6%	81.8%

Table 4. Speech recognition accuracy.

Speech Recognition	MFCC+HMM [8]	MTDC+GMM [30]	DNN	LDA2FNN
Average correct rate	76.6%	77.8%	76.0%	81.8%

4.3 Theoretical Analysis

Table 5 summarizes the analysis of different fuzzy classifiers. In fact, all classifiers have the same cost function $E = \frac{1}{2}[y - y^d]^2$, but they differ in their second cost function. Therefore, the theoretical analysis focuses on the effect of the second cost function. In SONFIN-PCA, the cost function of $\arg\max_e [e^T \Sigma_X e\ ]$ can increases the variance. In MD-SOFN-LDA, the cost function of $\arg\max_e \frac{e^T S_B e}{e^T S_W e}$ maximizes the between-class variance and minimizes the within-class variance. Compared with SONFIN-PCA, MDSOFN-LDA seeks directions that maximize the discriminability instead of maximizing the variance, explaining why MDSOFN-LDA are better than SONFIN-PCA. To consider the cost function of LDA2FNN, it does not only consider $\arg\max_e \frac{e^T S_B e}{e^T S_W e}$ but also adopt interval type-2 fuzzy-sets to model the uncertainty. Hence, LDA-based type-2 fuzzy rules yielded the most discriminative representation in the experiments.

Table 5. Analysis of different fuzzy neural networks.

	SONFIN-PCA	MDSOFN-LDA	SEIT2FNN	LDA2FNN
Fuzzy Type	Type-1 fuzzy set	Type-1 fuzzy set	Type-2 fuzzy set	Type-2 fuzzy set
Consequent Parts	TS-type	TS-type	TS-type	TS-type
Cost Functions	1. $E = \frac{1}{2}\left[y - y^d\right]^2$ 2. $\arg\max_e [e^T \Sigma_X e\ ]$	1. $E = \frac{1}{2}\left[y - y^d\right]^2$ 2. $\arg\max_e \frac{e^T S_B e}{e^T S_W e}$	1. $E = \frac{1}{2}\left[y - y^d\right]^2$ 2. None	1. $E = \frac{1}{2}\left[y - y^d\right]^2$ 2. $\arg\max_e \frac{e^T S_B e}{e^T S_W e}$
Characteristic	1. PCA maximizes the covariance. 2. The gradient descent method adjusts the Gaussian function.	1. LDA maximizes between-class and minimizes within-class. 2. The gradient descent method adjusts the Gaussian function.	In handing problems with uncertainties such as noisy data, type-2 fuzzy-systems usually outperform their type-1 counterparts.	The antecedent parts adopt interval type-2 fuzzy-sets to model the uncertainty, and the consequent parts adopt LDA to enhance the discriminability.
Drawback	If LDA and PCA have the same between-class, PCA increases the variance and decreases the discriminative capabilities.	Need the information of between-class and within-class.	The cost function does not consider the discriminability.	Need the information of between-class and within-class.

5. CONCLUSIONS

Two important classification problems in speech signal processing are speech detection and recognition. They both are very crucial to human-machine interaction. However, they are easily affected by noisy environments. Hence, a linear-discriminant-analysis-based type-2 fuzzy-neural-network (LDA2FNN) is proposed to overcome the problem of uncertainties in noisy speech classification. Interval type-2 fuzzy-sets are adopted in the antecedent parts to model and minimize the effect of uncertainty. Linear-discriminant-analysis

(LDA) is then applied in the consequent parts to increase the discriminability. The novelty of LDA2FNN is its consideration of both uncertainty and discriminability. Besides, LDA2FNN can classify noisy data, while preserving a small fuzzy network size. The effectiveness of the proposed LDA2FNN is demonstrated by testing speech detection and recognition. Experimental results and theoretical analysis show that the proposed LDA2FNN performs better other fuzzy neural networks.

REFERENCES

1. J. C. Lv, K. K. Tan, Z. Yi, and S. Huang, "A family of fuzzy learning algorithms for robust principal component analysis neural networks," *IEEE Transactions on Fuzzy Systems*, Vol. 18, 2010, pp. 217-226.
2. K. Honda, A. Notsu, and H. Ichihashi, "Fuzzy PCA-guided robust *k*-means clustering," *IEEE Transactions on Fuzzy Systems*, Vol. 18, 2010, pp. 67-79.
3. C. F. Juang and C. T. Lin, "An on-line self-constructing neural fuzzy inference network and its applications," *IEEE Transactions on Fuzzy Systems*, Vol. 16, 1998, pp. 12-32.
4. G. D. Wu and C. T. Lin, "Word boundary detection with mel-scale frequency bank in noisy environment," *IEEE Transactions on Audio*, *Speech and Language Processing*, Vol. 8, 2000, pp. 541-554.
5. G. D. Wu and P. H. Huang, "A maximizing-discriminability-based self-organizing fuzzy network for classification problems," *IEEE Transactions on Fuzzy Systems*, Vol. 18, 2010, pp. 362-373.
6. W. Y. Cheng and C. F. Juang, "An incremental support vector machine-trained TS-type fuzzy system for on-line classification problems," *Fuzzy Sets and Systems*, Vol. 163, 2011, pp. 24-44.
7. A. Biem, "Minimum classification error training for online handwriting recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 28, 2006, pp. 1041-1051.
8. J. W. Hung and L. S. Lee, "Optimization of temporal filters for constructing robust features in speech recognition," *IEEE Transactions on Audio*, *Speech and Language Processing*, Vol. 14, 2006, pp. 808-832.
9. M. A. Shoorehdeli, M. Teshnehlab, and A. K. Sedigh, "Training ANFIS as an identifier with intelligent hybrid stable learning algorithm based on particle swarm optimization and extended Kalman filter," *Fuzzy Sets and Systems*, Vol. 160, 2009, pp. 922-948.
10. C. F. Juang, Y. C. Chang, and C. M. Hsiao, "Evolving gaits of a hexapod robot by recurrent neural networks with symbiotic species-based particle swarm optimization," *IEEE Transactions on Industrial Electronics*, Vol. 58, 2011, pp. 3110-3119.
11. C. F. Juang and Y. C. Chang, "Evolutionary group-based particle swarm-optimized fuzzy controller with application to mobile robot navigation in unknown environments," *IEEE Transactions on Fuzzy Systems*, Vol. 19, 2011, pp. 379-392.
12. C. F. Juang, T. L. Jeng, and Y. C. Chang, "An interpretable fuzzy system learned through online rule generation and multi-objective ACO with a mobile robot control application," *IEEE Transactions on Cybernetics*, Vol. 46, 2016, pp. 2706-2718.

13. C. F. Juang, C. W. Hung, and C. H. Hsu, "Rule-based cooperative continuous ant colony optimization to improve the accuracy of fuzzy system design," *IEEE Transactions on Fuzzy Systems*, Vol. 22, 2014, pp. 723-735.
14. C. F. Juang and K. J. Juang, "Circuit implementation of a TSK-type interval type-2 neural fuzzy system with on-chip parameter tuning ability," *IEEE Transactions on Industrial Electronics*, Vol. 64, 2017, pp. 4266-4275.
15. C. F. Juang and P. H. Wang, "An interval type-2 neural fuzzy classifier learned through soft margin minimization and its human posture classification application," *IEEE Transactions on Fuzzy Systems*, Vol. 23, 2015, pp. 1474-1487.
16. C. F. Juang, W. Y. Cheng, and C. W. Liang, "Speedup of learning in interval type-2 neural fuzzy systems through graphic processing units," *IEEE Transactions on Fuzzy Systems*, Vol. 23, 2015, pp. 1286-1298.
17. C. F. Juang and W. S. Jang, "A type-2 neural fuzzy system learned through type-1 fuzzy rules and its FPGA-based hardware implementation," *Applied Soft Computing*, Vol. 18, 2014, pp. 302-313.
18. W. Y. Cheng and C. F. Juang, "A fuzzy model with online incremental SVM and margin-selective gradient descent learning for classification problems," *IEEE Transactions on Fuzzy Systems*, Vol. 22, 2014, pp. 324-337.
19. G. C. Chen and C. F. Juang, "An accuracy-oriented self-splitting fuzzy classifier with support vector learning in high-order expanded consequent space," *Applied Soft Computing*, Vol. 15, 2014, pp. 231-242.
20. J. Markoff, *Scientists See Promise in Deep-Learning Programs*, New York Times, Nov 24, 2012.
21. J. Huang and B. Kingsbury, "Audio-visual deep learning for noise robust speech recognition," in *Proceedings of IEEE International Conference on Acoustics*, *Speech and Signal Processing*, 2013, pp. 7596-7599.
22. A. Mohamed, G. Dahl, and G. Hinton, "Acoustic modeling using deep belief networks," *IEEE Transactions on Audio*, *Speech*, *and Language Processing*, Vol. 20, 2012, pp. 14-22.
23. N. Morgan, "Deep and wide: Multiple layers in automatic speech recognition," *IEEE Transactions on Audio*, *Speech*, *and Language Processing*, Vol. 20, 2012, pp. 7-13.
24. J. M. Mendel, "Computing derivatives in interval type-2 fuzzy logic systems," *IEEE Transactions on Fuzzy Systems*, Vol. 12, 2004, pp. 84-98.
25. C. F. Juang and Y. W. Tsao, "A self-evolving interval type-2 fuzzy neural network with online structure and parameter learning," *IEEE Transactions on Fuzzy Systems*, Vol. 16, 2008, pp. 1411-1424.
26. G. D. Wu and C. T. Lin, "A recurrent neural fuzzy network for word boundary detection in variable noise-level environments," *IEEE Transactions on Systems*, *Man*, *and Cybernetics*, Vol. 31, 2001, pp. 84-97.
27. M. G. Varga and H. J. M. Steeneken, "Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Computer Speech & Language*, Vol. 12, 1993, pp. 247-251.
28. K. C. Wang, "A novel approach based on adaptive long-term sub-band entropy and multi-thresholding scheme for detecting speech signal," *IEICE Transactions on Information and Systems*, Vol. E95-D, 2012, pp. 2732-2736.

29. C. C. Tu and C. F. Juang, "Recurrent type-2 fuzzy neural network using Haar wavelet energy and entropy features for speech detection in noisy environments," *Expert Systems with Applications*, Vol. 39, 2012, pp. 2479-2488.
30. C. T. Lin, H. W. Nein, and J. Y. Hwu, "GA-based noisy speech recognition using two-dimensional cepstrum," *IEEE Transactions on Audio*, *Speech*, *and Language Processing*, Vol. 8, 2000, pp. 664-675.

Gin-Der Wu (吳俊德) received the B.S. degree in Engineering Science from the National Cheng-Kung University, Tainan, Taiwan, in 1996 and the Ph.D. degree in Electrical and Control Engineering from the National Chiaog-Tung University, Hsinchu, Taiwan, in 2000. In 2002, he joined VIA Technologies, INC., as a senior engineer in Chipset R&D Division. In 2003, he joined ALI Corporation, as a senior engineer in Digital AV Product Business Division. Since 2004, he has been with the Department of Electrical Engineering, National Chi Nan University, Nantou, Taiwan, where he became a Full Professor in 2014. His current research interests include fuzzy neural networks, field-programmable-gate-array chip design, and digital signal processsing.

Zhen-Wei Zhu (朱振緯) received the Master degree and the Ph.D. degree in Electrical Engineering from the National Chi Nan University, Nantou, Taiwan. His current research interests include VLSI, DSP architecture and fuzzy neural networks.