Available online at www.sciencedirect.com
ScienceDirect
Neurocomputing 71 (2007) 121–130
NEUROCOMPUTING
www.elsevier.com/locate/neucom

Birdsong recognition using prediction-based recurrent neural fuzzy networks

Chia-Feng Juang*, Tai-Mou Chen

Department of Electrical Engineering, National Chung-Hsing University, Taichung 402, Taiwan, ROC

Available online 30 August 2007

Abstract

Automatic birdsong recognition using prediction-based singleton-type recurrent neural fuzzy networks (SRNFNs) is proposed in this paper. The recognition task consists of two stages. The first stage segments a significant portion from a birdsong sequence and the second stage performs recognition. For birdsong segmentation, an easy but effective segmentation approach based on time domain energy is proposed. For recognition, the linear predictive coding (LPC) coefficients of each frame in a segmented birdsong are extracted and used as features. These features are fed as inputs to SRNFN recognizers. The SRNFN is constructed by recurrent fuzzy if–then rules with fuzzy singletons in the consequences, and its recurrent aspect makes it suitable for processing patterns with temporal characteristics. In birdsong recognition, the sample prediction technique is used, where one SRNFN is responsible for learning the temporal birdsong relationships of only one species. The prediction error of each SRNFN is then used as a criterion for recognition. Experiments with 10 species of birds and their songs are performed, and a high recognition rate is achieved. Comparisons with a Takagi–Sugeno–Kang (TSK)-type recurrent fuzzy network (TRFN) and backpropagation neural network are also made in the experiments.

Keywords: Bird species identification; Temporal sequence prediction; Neural networks; Recurrent fuzzy systems; LPC coefficients

1. Introduction

Automatic bird species identification using computers or custom dedicated hardware enables the automatic identification of birds without the inquiry of experts. Bird species may be identified by their physical features or sounds. In many areas, bird sounds can be heard but it is very difficult to actually see the birds. In these cases, birdsong recognition is important. Automatic birdsong recognition technology can help biologists monitor bird migration and population in the field. A bird monitoring system with bird classification capacities may help avoid bird strikes, i.e., birds hitting airplanes [8]. For tourist industry applications, this technology is helpful to people who practice bird-watching as a hobby. Interest in this technology has risen recently and some automatic birdsong recognition approaches have been proposed [1,4,5,8,11]. Automatic bird species identification based on sinusoidal modeling of syllables was proposed in [8]. This is a fairly simple approach without any intelligent or context-aware processing tasks for a number of species. However, recognition results are only good for some species of birds. In [11], linear predictive coding followed by windowed Fourier transform was used to extract recognition feature vectors. These feature vectors were fed to a backpropagation neural network (BPNN) for recognition.

Bird sounds are composed of temporal patterns. For a temporal sequence, the value of one output pattern depends not only on present input, but also on the preceding or following inputs. If a feedforward network is used for this task, the number of delayed inputs and outputs must be known in advance. The problem of this approach is that the exact order of the temporal system is usually unknown. The usage of a long tapped delay input increases input dimensions and creates a large network. In light of these problems, a recurrent network is more suitable than a feedforward network [6]. Some recurrent neural fuzzy networks have already been proposed [6,7,9,10,12,15,17,19]. One category includes external

*Corresponding author. Tel.: +886 4 2284 0688; fax: +886 4 2285 1410.
E-mail address: cfjuang@dragon.nchu.edu.tw (C.-F. Juang).

0925-2312/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2007.08.011

feedback as recurrence [10,12,15,19]. The disadvantage of this temporal sequence processing approach is that the number of adjacent patterns included in the recurrent model must be known in advance. Another recurrent fuzzy network category includes fuzzy models that do not use adjacent input patterns [6,7,9,17]. In [17], an output-recurrent fuzzy neural network was proposed where the output values are fed back as input values. For prediction problems, this network functions similarly to a feedforward neural fuzzy network with the addition of past inputs. In [6], a Takagi–Sugeno–Kang (TSK)-type recurrent fuzzy network (TRFN) was proposed where the TSK-type consequence is a linear combination of input variables plus a constant. The performance of the TRFN was higher than other recurrent networks compared in [6]. This paper proposes a recognizer design based on a recurrent fuzzy network. In addition, a singleton-type recurrent fuzzy neural network (SRNFN) is proposed which simplifies the TRFN consequence to a fuzzy singleton. For complex temporal mapping problems, it is necessary to use a TRFN for high mapping accuracy. However, for simple temporal mapping problems like birdsong recognition, the required number of rules is small and an SRNFN is sufficient. Since the recognizers designed in this paper have high dimensional inputs and outputs, experiments in Section 4 show that an SRNFN can achieve a recognition rate similar to a TRFN recognition rate. However, the number of parameters in a TRFN is much greater than in an SRNFN. This is due to the use of TSK type consequences in a TRFN.

An SRNFN performs birdsong recognition by sample prediction instead of classification. In the classification approach, the number of output nodes in a neural network is usually equal to the number of classes to be recognized. If the number of classes increases, then the scaling problem in neural network training occurs. As a result, it becomes very difficult to train a single network to perform classification. The BPNN recognizer proposed in [11] uses this classification approach. To solve this scaling problem, a divide-and-conquer technique is used in the sample prediction-based approach. In [14], human speech recognition was proposed using prediction error. However, feedforward neural networks were used as predictors in this study. In this paper, predictors are designed using SRNFNs. This helps ease the time-consuming network input dimension task and reduce network size. An automatic birdsong segmentation and recognition experiment was conducted using prediction-based SRNFNs. In the experiment, 10 species of birds were identified by their songs. A high recognition rate was achieved in the experiment. Comparisons with BPNN classification and prediction-based TRFN recognition approaches were made to verify SRNFN recognizer efficiency.

This paper is organized as follows. Section 2 introduces the birdsong segmentation approach and the features used for recognition. In Section 3, the SRNFN structure is introduced. Recognition configuration of a prediction-based SRNFN is also introduced in this section. Section 4 describes the experimental process. Conclusions are made in Section 5.

2. Birdsong segmentation and feature extraction

In this study, 10 species of birds that can be found in Taiwan were identified by song recognition. Their scientific names, common names, and sound waveforms are shown in Fig. 1. The birdsongs in Fig. 1 show that there may be many pauses in a birdsong. Bursts of sound usually come between two perceptible pauses, and a birdsong sequence usually contains many bursts. The burst durations are different for different birds, and for some birds these bursts are not clearly perceptible. In this paper, energy parameters were used to segment significant portions from a birdsong sequence. The birdsongs were recorded at 44.1 kHz sampling rate. For a birdsong sequence, the time domain energy of each non-overlapping frame was computed, and each frame was 512 samples long. The maximum energy in the sequence was then found. Starting from the middle sample of the maximum energy frame, the succeeding and preceding 25 600 samples were segmented. This created a constant length sequence with 51 201 samples segmented. A constant sequence length was used so that pause sequence was also segmented for birds with small burst duration. On the contrary, for a bird with long burst duration, only the burst sequence was segmented for segmentation. No energy thresholds are required to set a priori before segmentation. This is in contrast to clean speech segmentation by time domain energy, where segmentation is based on some energy thresholds and only speech is segmented. The proposed method is useful in clean or high signal-to-noise environment. Like noisy speech detection problems, automatic segmentation of noisy birdsong with low signal-to-noise ratio is not easy and will be studied in the feature. For illustration, two birdsongs, their time domain energies, and segmented sequences are shown in Fig. 2.

Features for birdsong recognition can be extracted from birdsong analysis [1,4,5,16] or from the way a birdsong is generated [11]. For the birdsong analysis approach, it was stated in [2] that birdsongs are typically divided into four levels: notes, syllables, phrases, and song. Birdsong recognition can be performed according to variances on one or more of these levels. Among these levels, syllables are the most elementary building blocks and are more suitable for birdsong recognition [1,5]. In [5], birdsong recognition is performed using syllables but the results are good only for some species of birds. For the feature extraction approach based on birdsong generation, it was found that bird sounds are generated mainly by the syrinx. Humans generate speech by exciting the vocal cords. The way birds generate sound is similar to the way humans generate speech. Thus, techniques used for human speech recognition may possibly be applied to birdsong recognition. For voiced human speech, the all-pole model of linear

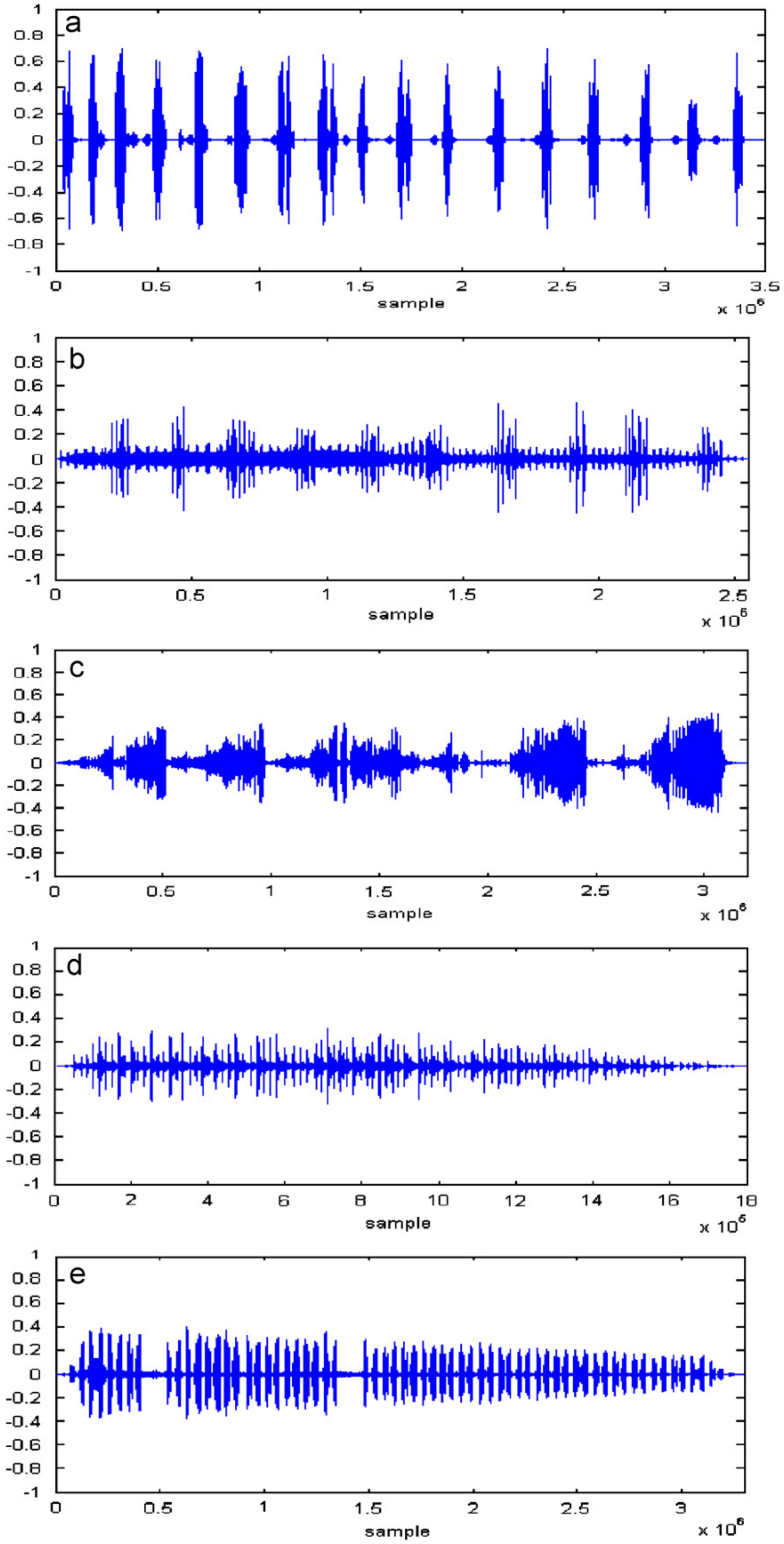

Fig. 1. The names and sound waveforms of 10 species of birds studied in this paper: (a) scientific name: *Brachypteryx montana*, trivial name: Blue Shortwing, (b) scientific name: *Pycnonotus taivanus*, trivial name: Taiwan Bulbul, (c) scientific name: *Garrulax canorus*, trivial name: Hwa-Mei, (d) scientific name: *Hypsipetes madagascariensis*, trivial name: Black Bulbul, (e) scientific name: *Megalaima oorti*, trivial name: Muller's Barbet, (f) scientific name: *Podiceps ruficollis*, trivial name: little Grebe, (g) scientific name: *Otus elegans botelensis*, trivial name: Lanyu Scops Owl, (h) scientific name: *Ninox scutulata* (*Raffles*), trivial name: Brown Hawk Owl, (i) scientific name: *Amaurornis phoenicurus*, trivial name: White-breasted Water Hen, (j) scientific name: *Phasianus colchicus*, trivial name: Ring-necked Pheasant.

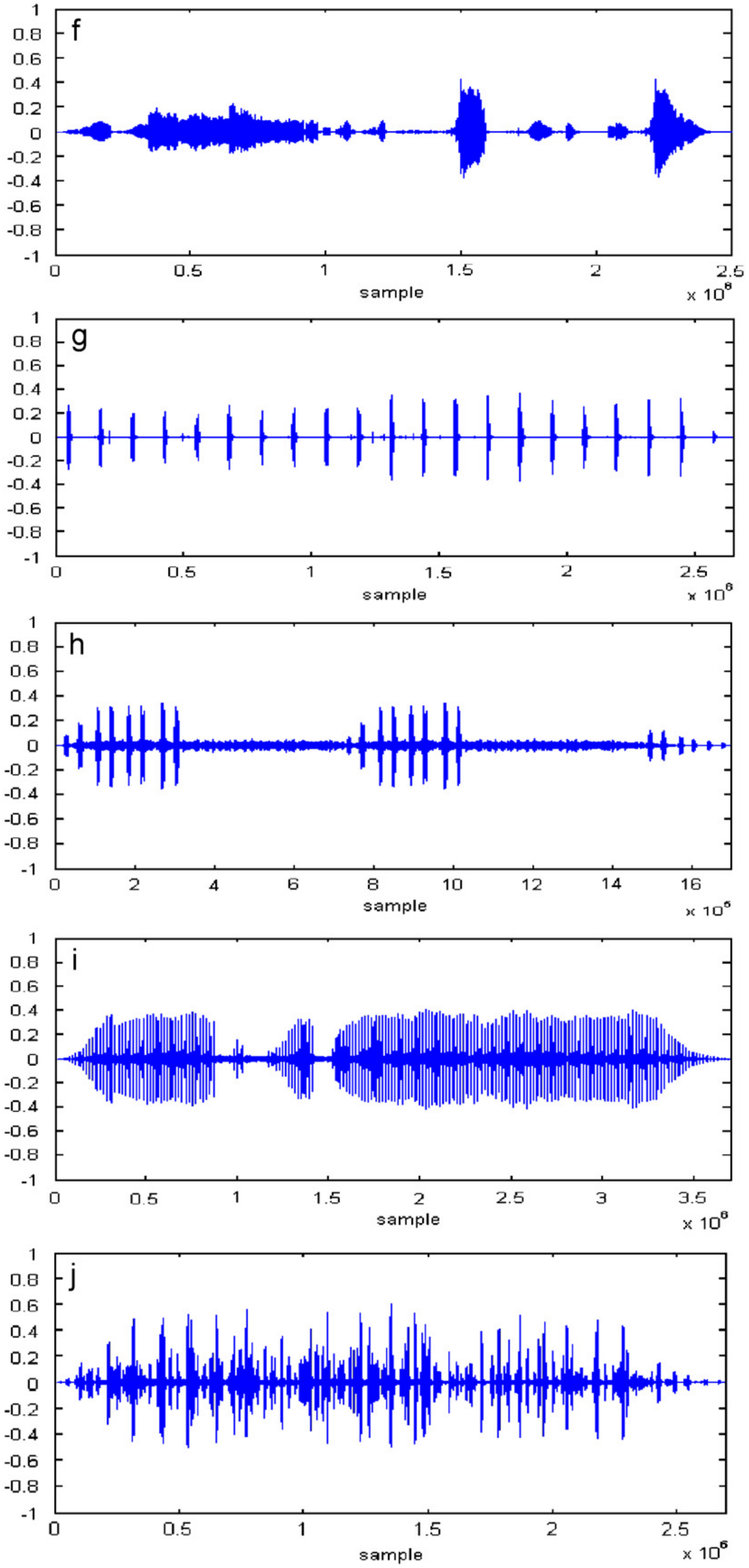

Fig. 1. (*Continued*)

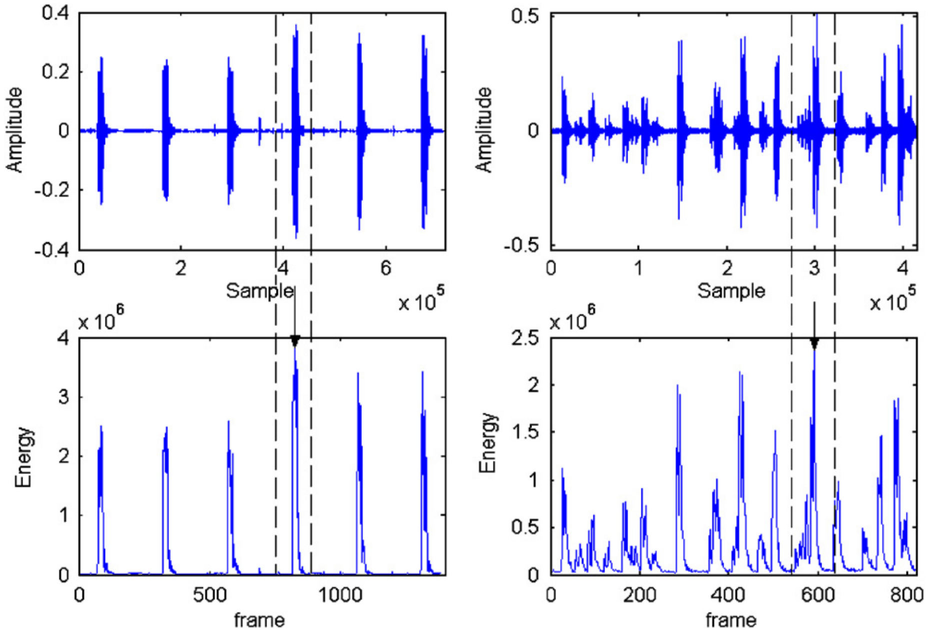

Fig. 2. Waveforms of two birdsong sequences, their time-domain energies, and segmented birdsongs.

predictive coding (LPC) provides a good approximation of the vocal track spectral envelope. The basic idea behind the LPC model is that a given speech sample at time n, $s(n)$, can be approximated as a line combination of the past p speech samples, such that

$$s(n) = \sum_{i=1}^{p} a_i s(n-i) + Gu(n), \tag{1}$$

where $u(n)$ is a normalized excitation and G is the gain of the excitation. Parameters $a_1,\ldots,a_p$ are called LPC coefficients and p is called the prediction order. The LPC coefficients can be found by autocorrelation analysis and Durbin's method [13]. LPC coefficients had been used for speech recognition [13]. For birdsong recognition in [11], fifteen order LPC coefficients were computed for each birdsong frame. Then, fast Fourier transform of the whitening LPC coefficients was used to produce nine unique spectral magnitudes. These nine spectral magnitudes together with song length were then used as recognition features. In this paper, LPC coefficients were used directly as birdsong recognition features. Twelve order LPC coefficients were used in this paper. Experiments using fifteen order LPC coefficients were also conducted and the performance is similar to that using twelve orders, so twelve orders were selected to reduce recognizer input dimension. Experiments also showed that for prediction-based SRNFN recognizers, LPC coefficients achieve high recognition results. Thus, no additional Fourier transform calculations were required as in [11]. Framing was performed using an overlapping Hamming window with 512 samples, where the overlapping width is half a frame. That is, the length of each frame is 512 samples.

3. Recognition using prediction-based SRNFN

The proposed SRNFN is constructed from a series of recurrent fuzzy if–then rules with fuzzy singletons in the consequences. In SRNFN, there are initially no rules, and the rules are constructed by on-line structure and parameter learning. That is, no pre-assignment of fuzzy rules in SRNFN is required. With flexibility of partition in the precondition part, and feedback structure, SRNFN has the admirable property of small network size and high learning accuracy. In addition, the use of singleton-type consequences makes SRNFN a more efficient recognizer in birdsong recognition problems than the TRFN [6] that uses TSK-type consequences. In this section, the structure of SRNFN is introduced followed by recognition by prediction-based SRNFN.

3.1. Structure of SRNFN

The structure of SRNFN is shown in Fig. 3. A network with two external inputs and a single output is considered here for convenience. In contrast to six-layered TRFN, there are only five layers in SRNFN. This five-layered network realizes a recurrent fuzzy network of the following form:

Rule 1: IF $x_1(t)$ is A_{11} and $x_2(t)$ is A_{12} and $h_1(t)$ is G THEN $y(t+1)$ is b_1 and $h_1(t+1)$ is w_{11} and $h_2(t+1)$ is w_{21},

Rule 2: IF $x_1(t)$ is A_{21} and $x_2(t)$ is A_{22} and $h_2(t)$ is G THEN $y(t+1)$ is b_2 and $h_1(t+1)$ is w_{12} and $h_2(t+1)$ is w_{22},

where A_{ij} and G are fuzzy sets, w_{ij} and b_i are fuzzy singleton values functioned as the consequent parameters for

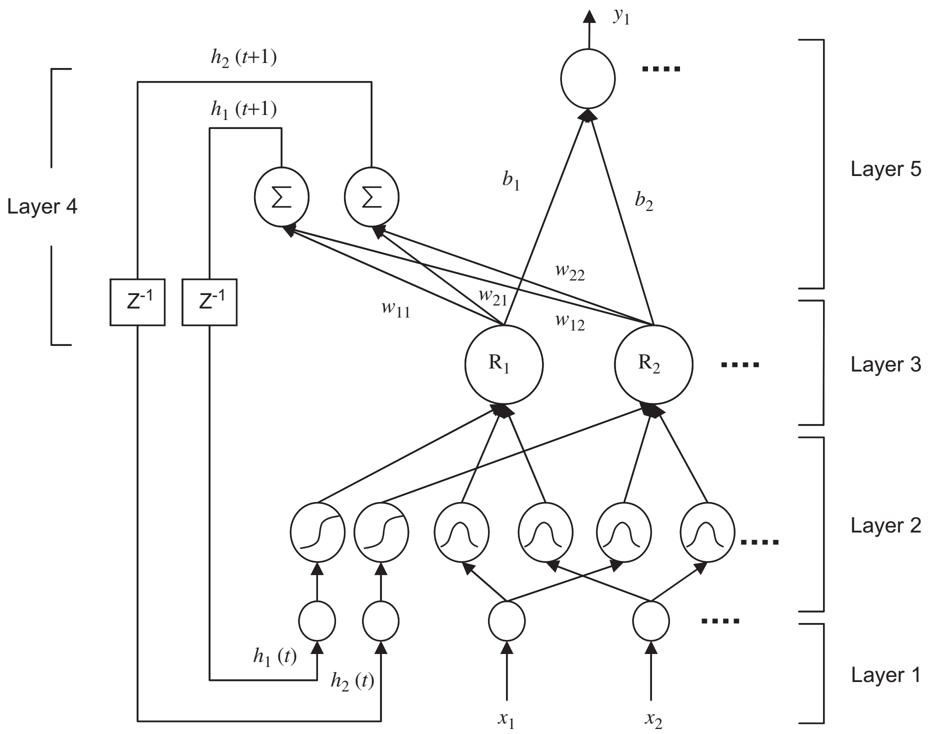

Fig. 3. Structure of the singleton-type recurrent neural fuzzy network (SRNFN).

inference output h_i and y, respectively. In Fig. 3, a network constructed by the above two rules is shown. There are two external input variables x_1 and x_2, and single output y. Accordingly, SRNFN has two nodes in layer 1 and one node in layer 5. To give a clear understanding of the mathematical function of each node, functions of SRNFN are described layer by layer below.

Nodes in layer 1 are input nodes. The node only transmits external input values x_j to layer 2. Nodes in layer 2 are called input term nodes and act as membership functions to express the input fuzzy linguistic variables. Two types of membership functions are used in this layer. Membership functions on external variables are constructed according to locally spatial mapping property. For external input x_j, Gaussian membership functions which locally maps the input spatial space to the output space are used, and the mathematical function is

$$\mu_i(x_j) = \exp\left\{-\frac{(x_i - m_{ij})^2}{\sigma_{ij}^2}\right\}, \tag{2}$$

where m_{ij} and σ_{ij} are, respectively, the center and the width of the Gaussian membership function. For internal variable h_i, the following sigmoid membership function is used:

$$\mu^s(h_i) = \frac{1}{1 + \exp\{-h_i\}}. \tag{3}$$

Each internal variable has a single corresponding fuzzy set. Each node in layer 3 is called a rule node. The number of rule nodes in this layer is equal to the number of fuzzy sets corresponding to each external linguistic input variable. For example, in Fig. 3, if there are two rule nodes in layer 3, then there are also two fuzzy sets in inputs x_1 and x_2. The output of each node in this layer is determined by fuzzy AND operation. Here, the product operation is utilized to determine the firing strength of each rule. The function of each rule is

$$\Phi_i = \mu^s(h_i)\prod_{j=1}^{n}\mu_i(x_j) = \frac{1}{1 + \exp\{-h_i\}} \times \exp\left\{-\left(\sum_{j=1}^{n}\frac{(x_j - m_{ij})^2}{\sigma_{ij}^2}\right)\right\}, \quad i = 1, \ldots, r, \tag{4}$$

where r is the number of fuzzy rules and is the number of external inputs. Nodes in layer 4 are called context nodes and perform defuzzification operation for internal variables h. The number of internal variables in this layer is equal to the rule nodes. The link weights represent the singleton values in the consequent part of the internal rules. The simple weighted sum is calculated in each node:

$$h_i = \sum_{k=1}^{r} w_{ik}\Phi_k. \tag{5}$$

As in Fig. 3, the delayed value of h_i is fed back to layer 1 and acts as an input variable to the precondition part of a rule. Each rule has a corresponding internal variable h and is used to decide the influence degree of temporal history to the current rule. Nodes in layer 5 are called defuzzification nodes. Each node performs weighted average operation for output y. The node in this layer computes the output signal y of the SRNFN. The output node together with links connected to it acts as a defuzzifier. The mathematical function is

$$y = \frac{\sum_{k=1}^{r} b_k \Phi_k}{\sum_{k=1}^{r} \Phi_k}. \tag{6}$$

3.2. Recognition using prediction-based SRNFN

For pattern recognition, neural networks are usually employed as pattern discriminators, where each node in the output layer represents one class name. Here, SRNFNs are used as pattern predictors instead of pattern discriminators. To recognize n classes of birdsongs, n SRNFNs are created. One SRNFN is used to learn the temporal relationship of one class of birdsong. This approach can be regarded as a kind of divide-and-conquer technique and it helps to avoid the scaling problem encountered in training neural networks. As the number of birdsong classes to be recognized increases, one only needs to add new SRNFNs, while the original trained SRNFNs are kept unchanged. Each SRNFN takes one frame feature from the birdsong and attempts to predict the following one. Only one frame is fed as network input since the SRNFN has the ability to learn the temporal relationships between current data and the past history of evens. If feedforward neural networks are used, then there is the problem of how many frames should be selected as network inputs. This selection task is eased in design of SRNFN as only one frame is fed as input. Furthermore, a less input dimension may reduce the network size.

The recognition task of SRNFN recognizer consists of two phases: the training phase and the test phase. During training, frame features in a birdsong are presented to the corresponding SRNFN in sequence. The SRNFN takes the current frame feature and attempts to predict the next one. The predicted frame feature is compared with the actual one, and the prediction error is used to train SRNFN. The task of constructing the SRNFN is divided into two subtasks: structure learning and parameter learning, which are both performed concurrently. That is, for each input data, the structure learning is performed followed by parameter learning. There are no rules initially in SRNFN, and the objective of the structure learning is to decide the number of fuzzy rules, initial location of membership functions, and initial consequent parameters. Before structure learning, two parameters, F_{in} and β should be assigned in advance. The parameter F_{in} is a threshold in (0, 1) that influences the total number of rules generated. A larger value of F_{in} will generate a larger number of rules. The parameter β decides the overlap degree between two rules (clusters). That is, it decides the initial width of each Gaussian fuzzy set. The criterion of generating a new rule is the same as that used in TRFN, where the spatial firing strength $F_i = \prod_{j=1}^{n} \mu_i(x_j)$ in layer 3 is used as the criterion to decide if a new fuzzy rule should be generated. For each incoming data $\vec{x}(t)$, find

$$I = \arg \max_{1 \leqslant i \leqslant r(t)} F_i(\vec{x}(t)), \tag{7}$$

where $r(t)$ is the number of existing rules at time t. If $F_{\text{I}} < F_{\text{in}}$, then a new rule is generated. Once a new rule is generated, the initial centers and widths of the corresponding membership functions are computed by

$$m_{(r(t)+1)i} = x_i(t) \quad \text{and} \quad \sigma_{(r(t)+1)i} = \beta \exp\left\{ -\left(\sum_{j=1}^{n} \frac{(x_j - m_{Ij})^2}{\sigma_{Ij}^2} \right)\right., \quad i = 1, \ldots, n, \tag{8}$$

respectively. The number of fuzzy sets in each external input dimension is equal to the number of fuzzy rules. Generation of a context node in layer 4 accompanies the generation of a rule. As to the parameter learning, if the cost function is

$$E(t+1) = \tfrac{1}{2}(y(t+1) - y^{\text{d}}(t+1))^2, \tag{9}$$

then consequent parameters b_i, $i = 1,\ldots,r$ are updated as follows:

$$b_i(t+1) = b_i(t) - \eta \frac{\partial E(t+1)}{\partial b_i(t)}, \tag{10}$$

where

$$\frac{\partial E(t+1)}{\partial b_i} = (y(t+1) - y^{\text{d}}(t+1)) \frac{\Phi_i}{\sum_{k=1}^{r} \Phi_k} \tag{11}$$

and the other parameters are tuned by real-time recurrent learning algorithm [18].

For demonstration of SRNFN training, it is unfolded in time domain, as shown in Fig. 4, where feature vector in frame t is denoted by $\vec{s}(t)$. If feature vector $\vec{s}(t)$ is fed as SRNFN input, the corresponding desired output $\vec{y}^{\,\text{d}}(t+1)$ is $\vec{s}(t+1)$, i.e., feature vector of the next frame. Fig. 4 shows that SRNFN output $\vec{y}(t+1)$ depends not only on the current input $\vec{s}(t)$ but also on $h(t)$, a feature vector that records the temporal history of preceding birdsong patterns. At time t, the weight updating criterion is to minimize the prediction error $0.5\|\vec{y}(t+1) - \vec{y}^{\,\text{d}}(t+1)\|^2$. By real-time recurrent learning, update of the weights depends not only on the output error between $\vec{y}(t+1)$ and $\vec{y}^{\,d}(t+1)$, but also on the accumulated errors propagated from previous frames 1 to $t-1$. If the trained error is small, the SRNFN is considered to be a good model for the corresponding birdsong. Correspondingly, it will show poor prediction performance for other birdsongs. Based on this observation, an unknown birdsong is recognized based on the accumulated prediction errors.

After all classes of SRNFN are trained, the recognition task is switched to the test phase, where prediction error of each SRNFN is used as the recognition criterion. As shown in Fig. 5, when an unknown birdsong is presented, the predicted root mean squared error (RMSE) of each trained SRNFN is calculated and compared, and the SRNFN with the lowest RMSE is regarded as the category to which the birdsong belongs.

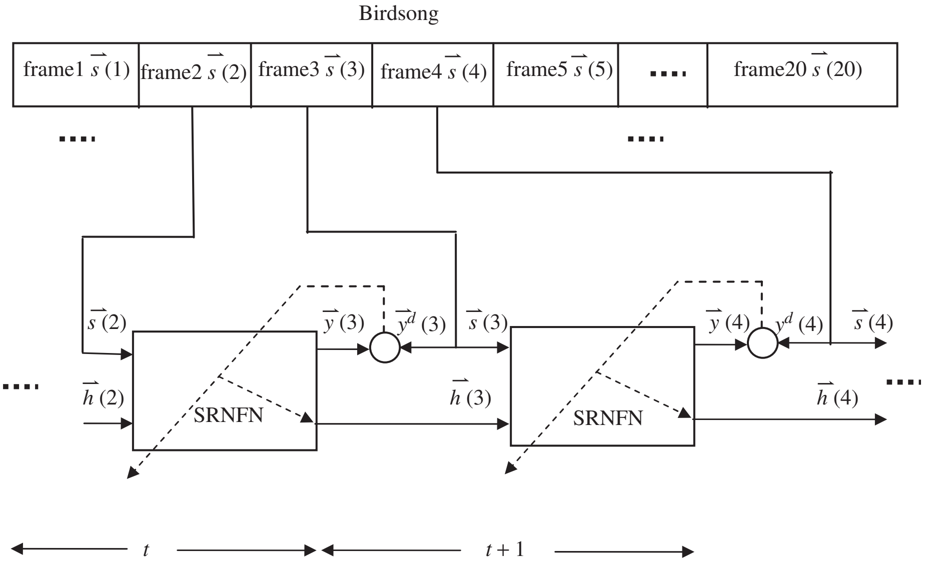

Fig. 4. Functional block diagram of SRNFN training.

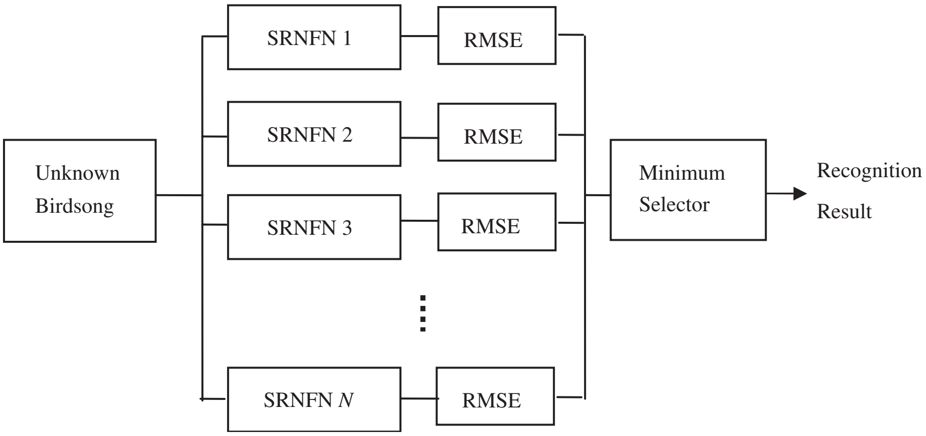

Fig. 5. Architecture of the SRNFN for birdsong recognition.

4. Experiments

The 10 species in Fig. 1 that can be found in Taiwan were recognized in the experiment. Birdsongs were obtained from commercial CD recordings [3]. Photos of these birds can also be found in the CD. The sampling rate was 44.1 kHz and each sample was represented by 16 bits. For recognizer training, 15 birdsongs of each bird species were collected. These birdsongs were manually segmented to 51 201 samples in length. Twelve order LPC coefficients of each overlapping frame in the manually segmented birdsongs were computed. Birdsong recognition using SRNFNs was experimentally tested, and 10 SRNFNs were created. Each SRNFN received the current frame features and predicted the next ones, creating 12 inputs and 12 outputs in each SRNFN. Each SRNFN recognizer was trained for 130 epochs with firing strength $F_{\text{in}} = 0.0001$, overlap parameter $\beta = 0.6$ and learning constant $\eta = 0.02$. After training, the number of fuzzy rules generated for each SRNFN ranged from two to three. That is, the proposed SRNFN was able to learn the temporal birdsong relationships with very few fuzzy rules. The exact number of rules in 10 SRNFN recognizers and their corresponding training results are shown in Table 1. For these training data, the overall recognition rate for 10 species was 98.67%. For recognizer testing, another 15 birdsong sequences of each species were collected. The birdsong segmentation method proposed in Section 2 was applied to these test sequences to segment significant birdsongs. Test results of the SRNFN recognizers are also shown in Table 1, where the recognition rate was 94.67%.

To determine the effectiveness of the proposed automatic birdsong segmentation method, experiments were also performed using manual segmentation of the test sequences. The manual segmentation recognition rate was 96.0%, and is shown in Table 2. These results show that recognition rates with automatic and manual segmentation methods are similar. That is, the proposed segmentation

Table 1
The number of rules in each SRNFN recognizer and recognition rates for both training and test data with birdsongs segmented automatically

Birds index	a	b	c	d	e	f	g	h	i	j	Average
Rule number	3	3	3	3	2	3	2	2	2	3	2.6 rules
Training rate (%)	100	100	100	93.33	100	100	100	100	93.33	100	98.67
Test rate (%)	93.33	100	100	86.67	93.33	100	100	73.33	100	100	94.67

Table 2
Recognition results of different recognizers for automatically and manually segmented birdsongs

Recognizer	SRNFN	TRFN [6]	BPNN [11]				
Input frame number	1	1	7	7	5	3	1
Rule/node number	26	26	16	8	10	15	30
Parameter number	720	4776	1536	768	720	720	720
Training epochs	130	130	1500	1500	1500	1500	1500
Training data (%)	98.67	99.33	93.0	80.4	89.19	81.75	70.19
Test data (manual) (%)	96.00	96.00	92.56	79.0	87.84	80.41	67.56
Test data (automatic) (%)	94.67	95.33	91.89	78.0	86.48	75.67	65.0

approach automates the recognition process with very little degradation in the recognition rate.

For evaluation, SRNFN performance is compared with a TRFN [6] and a BPNN [11]. For TRFN recognition, the same recognition configuration was used except that the SRNFN was replaced by the TRFN. With the same structure learning parameters, the numbers of fuzzy rules in 10 TRFNs are the same as those in 10 SRNFNs. Recognition results are shown in Table 2, and indicate that SRNFN performance is about the same as TRFN performance. However, Table 2 shows that the total number of parameters in 10 TRFNs is much greater than that in 10 SRNFNs. For each rule, there are 168 consequence parameters in the TRFN compared to only 12 consequent parameters in the SRNFN.

For a BPNN, one BPNN discriminator with 10 output nodes denoting 10 species of birds was used, as in the previous work [11]. In [11], the BPNN used only one frame feature as network inputs. Experimental results in Table 2 indicate a low recognition rate if only one frame feature is fed as input because the temporal information was not used. To incorporate temporal information, adjacent frame features were also fed as inputs. For a BPNN with five input frame features, the two succeeding and preceding frame features were fed as inputs in addition to the current frame feature, i.e., there were 60 inputs in all. The inputs shifted one frame at a time in a tapped line, and the parameter learning rate was 0.019. Different adjacent frame numbers were tested, and the results are shown in Table 2. In Table 2, the number of hidden nodes in BPNN was chosen so that the total BPNN parameter number was equal to that in the SRNFN for a fair comparison. In addition, for a BPNN with seven input frame features, two different hidden node numbers were tested. Results in Table 2 show that even though BPNN parameters were twice that of the SRNFN, its recognition rate was still lower than the SRNFN. The proposed SRNFN recognition configuration not only improves the recognition rate, but also avoids the task of input frame selection.

5. Conclusion

In this paper, a prediction-based SRNFN recognition approach is proposed for birdsong recognition. To automate the recognition approach, a simple but effective segmentation approach using time domain energy was used. With the proposed recognition configuration, a high recognition rate was achieved using simple linear predictive coding (LPC) coefficients. The proposed SRNFN is characterized by structure and parameter learning abilities which can generate different rule numbers automatically according to different training patterns. Thus, the rule numbers in 10 SRNFNs are not the same due to variances in different birdsongs. The performance of the proposed recognition approach is compared with prediction-based TRFN recognizers and a BPNN discriminator. Experimental results have verified the effectiveness of the SRNFN structure and the prediction-based recognition configuration. Recognition with a wider birdsong database, including a larger number of bird species and birdsongs from different birds of the same species, will be studied in the future. Moreover, since birdsongs in natural environments are usually corrupted by ambient sounds or other interferences, a robust recognizer for birdsong recognition in noisy environments will be studied in the future.

References

[1] S.E. Anderson, A.S. Dave, D. Margoliash, Template-based automatic recognition of birdsong syllables from continuous recordings, J. Acoust. Soc. Am. 100 (1996) 1209–1219.

[2] C.A. Catchpole, P.J.B. Slater, Bird Song: Biological Themes and Variations, Cambridge University Press, Cambridge, UK, 1995.

[3] CD title: Internet of birds, CD number: TCD-5032, Wind Records Co., Ltd., Taiwan, 1999.

[4] E.M. Date, R.E. Lemon, D.M. Weary, A.K. Richter, Species identity by birdsong: discrete or additive information?, Anim. Behav. 41 (1991) 111–120.

[5] A. Härmä, Automatic identification of bird species based on sinusoidal modeling of syllables, in: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, 2003, pp. 545–548.

[6] C.F. Juang, A TSK-type recurrent fuzzy network for dynamic systems processing by neural network and genetic algorithm, IEEE Trans. Fuzzy Syst. 10 (2) (2002) 155–170.

[7] C.F. Juang, S.J. Ku, A recurrent fuzzy network for fuzzy temporal sequence processing and gesture recognition, IEEE Trans. Syst. Man Cybern. 35 (3) (2005) 646–658.

[8] C. Kwan, et al., Bird song classification algorithms theory and experimental results, in: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, 2004, pp. 289–292.

[9] C.H. Lee, C.C. Teng, Identification and control of dynamic systems using recurrent fuzzy neural networks, IEEE Trans. Fuzzy Syst. 8 (4) (2000) 349–366.

[10] P.A. Mastorocostas, J.B. Theocharis, A recurrent fuzzy-neural model for dynamic system identification, IEEE Trans. Syst. Man Cybern. 32 (2) (2002) 176–190.

[11] A.L. McIlraith, H.C. Card, Birdsong recognition using backpropagation and multivariate statistics, IEEE Trans. Signal Process. 45 (11) (1997) 2740–2748.

[12] G.C. Mouzouri, J.M. Mendel, Dynamic nonsingleton fuzzy logic systems for nonlinear modeling, IEEE Trans. Fuzzy Syst. 5 (2) (1997) 199–208.

[13] L. Rabiner, B.H. Juang, Fundamentals of Speech Recognition, Prentice-Hall, Englewood Cliffs, NJ, 1993.

[14] J. Tebelskis, A. Waibel, Large vocabulary recognition using linked predictive neural networks, in: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, 1990, pp. 437–440.

[15] J.B. Theocharis, A high-order recurrent neuro-fuzzy system with internal dynamics: application to the adaptive noise cancellation, Fuzzy Sets Syst. 157 (4) (2006) 471–500.

[16] N.S. Thompson, K. LeDoux, K. Moody, A system for describing bird song units, Bioacoustics 5 (1994) 267–279.

[17] Y.C. Wang, C.J. Chien, C.C. Teng, Direct adaptive iterative learning control of nonlinear systems using an output-recurrent fuzzy neural network, IEEE Trans. Syst. Man Cybern. 34 (3) (2004) 1348–1359.

[18] R.J. Williams, D. Zipser, A learning algorithm for continually running recurrent network, Neural Comput. 1 (2) (1989) 270–280.

[19] J. Zhang, A.J. Morris, Recurrent neuro-fuzzy networks for nonlinear process modeling, IEEE Trans. Neural Networks 10 (2) (1999) 313–326.

Chia-Feng Juang received his B.S. and Ph.D. degrees in Control Engineering from the National Chiao-Tung University, Hsinchu, Taiwan, ROC, in 1993 and 1997, respectively.

From 1999 to 2001, he was an Assistant Professor of the Department of Electrical Engineering at the Chung Chou Institute of Technology. In 2001, he joined the National Chung Hsing University, Taichung, Taiwan, ROC, where he is currently a Professor of Electrical Engineering. He has authored and coauthored over 90 journal and conference papers in the area of computational intelligence. His current research interests are computational intelligence, intelligent control, computer vision, speech signal processing, and FPGA chip design. Over 300 journal papers have cited his research papers during the last 10 years (ISI citation database).

Tai-Mou Chen received his B.S. degree in Electrical Engineering from the National Chung-Hsing University, Taichung, Taiwan, ROC, in 2006. He is currently in military service. His research interests are speech signal processing and neural fuzzy systems.