Proceedings of International Joint Conference on Neural Networks, San Jose, California, USA, July 31 – August 5, 2011

Chord Recognition Using Neural Networks Based on Particle Swarm Optimization

Cheng-Jian Lin*, Chin-Ling Lee and Chun-Cheng Peng

Abstract—**A sequence of musical chords can facilitate musicians in music arrangement and accompaniment. To implement an intelligent system for chord recognition, in this paper we propose a novel approach using Artificial Neural Networks (ANN) trained by the Particle Swarm Optimization (PSO) technique and Backpropagation (BP) learning algorithm. All the training and testing data are generated from Musical Instrument Digital Interface (MIDI) symbolic data. Furthermore, in order to improve the recognition efficiency, the cadence is also included as an additional feature. Cadence is the structural punctuation of a melodic phrase and it is considered as an important feature for chord recognition. Experimental results of our proposed approach show that the addition of this feature improves significantly the recognition rate, and also that the ANN-PSO method outperforms ANN-BP in chord recognition. In addition, since preliminary experimental recognition rates are generally not stable enough, we further choose the optimal ANNs to propose a two-phase ANN model to ensemble the recognition results.**

Keywords*: *Chord Recognition, Cadence, Particle Swarm Optimization (PSO), Artificial Neural Network, MIDI.

I. Introduction

Music is a human expression in the medium of time using the structures of sounds or tones and silence. Common elements of music are melody, harmony, rhythm, dynamics, and the sonic qualities of timbre. The harmony is always described by chord symbols, while analyzing the overall harmonic structure of a piece of melody often starts with labeling every chord. A music chord is a set of simultaneous tones, and a sequence of chords (chord progression) can facilitate musicians in music arrangement and accompaniment.

The previous work on music structure analysis and the Music Information Retrieval field focused on music recording or symbolic music data, such as Musical Instrument Digital Interface (MIDI) data, which included music arrangement or accompaniment information. Also, some researchers worked on "automatic harmonization" or "chord generation" by using Hidden Markov Models (HMMs) to analyze musical pitches and chroma features. Dan, Ian and Sumit [1] developed "MySong", a system that uses a HMM to generate chords to accompany a vocal melody. They utilize the "Notes-only model" to choose chords for each measure by considering only the notes that appear in that measure. This model sampled the sequence of notes observed in a measure of melodies at regular intervals which are arbitrarily short and do not correspond to musical note durations, even if the note durations are an important feature for chord recognition. Kyogu and Malcolm [2] proposed an automatic chord recognition system from audio by using a supervised HMM. They used symbolic data such as MIDI in order to generate chord names and their boundaries as well as to create audio. It should be noticed that MIDI data and audio information already contained the harmonic and accompaniment notes which are generated by chords.

In this research, only successions of single tones in melody are considered and used to compose our training and testing datasets. Therefore, this paper focuses on the notes in a piece of melody and excludes harmonic and accompaniment information.

While a lot of Artificial Intelligent (AI) techniques have been developed to solve various complex engineering problems, Artificial Neural Networks (ANNs) have been used more frequently because of their advantages [3]. In the work of [4] a sequential neural network for harmonizing melodies in real-time was proposed. The network models aspects of human cognition and can be used as the basis for building an interactive system that automatically generates accompaniment for simple melodies in live performance situations. In addition, the neural network is able to learn the relations between important notes of the melody and their harmonies, which results to the ability of producing harmonies for new melodies in real-time basis. In this work, the ANN architectures that have been developed are trained by the Backpropagation (BP) algorithm and the Particle Swarm Optimization (PSO) technique. The datasets for training and testing our ANNs were extracted from the MIDI data of 84 children's ballads [5]. In order to verify the recognition ability of our proposed ANN-PSO, experimental results compared with the traditional ANN-BP indicate that our proposed approach outperforms the ANN-BP and single-phase ANN-PSO, in term of higher averaged recognition rates.

The rest of this paper is organized as follows. After the overview of ANN and the BP learning algorithm are presented in Section 2, Section 3 depicts the PSO technique. Before proposing our two-phase ANN model in Section 5, the

Cheng-Jian Lin is a Professor with the Department of Computer Science and Information Engineering, National Chin-Yi University of Technology, Taichung City, Taiwan 411, R.O.C. (phone: +886-4-2392-4505 ext. 8753; fax: +886-4-2391-7426; e-mail: cjlin@ncut.edu.tw).

Chin-Ling Lee is an Associate Professor with the Department of International Trade, National Taichung Institute of Technology, Taichung City, Taiwan 404, R.O.C.

Chun-Cheng Peng is a postdoctoral research associate with the Department of Computer Science and Information Engineering, National Chin-Yi University of Technology, Taichung City, Taiwan 411, R.O.C. (phone: +886-4-2392-4505 ext. 8753; fax: +886-4-2391-7426; e-mail: goudapeng@gmail.com).

*Correspondence author. Email:cjlin@ncut.edu.tw

preprocessing tasks for chord recognition are described in Section 4. Section 6 exhibits experimental results with further analysis. Then Section 7 draws conclusions.

II. An Overview of ANN and BP

An ANN is a parallel computational system consisting of one or many artificial neuron(s). In general, these artificial neurons are fully interconnected processing elements, just like the neurons in the human brain. By means of these specific connections between the artificial neurons, an ANN is capable of performing a particular task, such as pattern recognition or data classification. In this study, we apply a three-layered feedforward ANN as the main model. As shown in Figure 1, such an ANN has one input, one hidden and one output layers, with M input, K hidden and N output nodes in the corresponding layers, where the numbers of nodes, i.e., M, and N are application dependent, and K is structurally learnable.

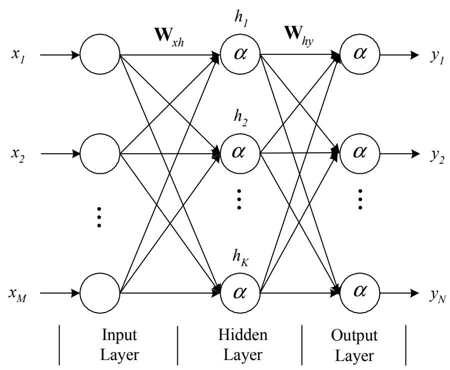


Figure 1. A simplified three-layer neural network.

Since any function with a given input-output mapping can be represented in the above ANN, learning and optimizing these weights, i.e., $\mathbf{W}_{xh}$ and $\mathbf{W}_{hy}$ in Figure 1, is the main issue in designing an ANN system. The standard BP algorithm is commonly adopted to tackle this problem, such as [6][7]. It is a powerful learning technique that enables the process to calculate the effective error between actual outputs and desired targets. In this fashion, the error signal can be propagated backward through the ANN to the synaptic connections of hidden layer. The weights of these synaptic connections can hence be adjusted to make actual outputs of the network as closer as possible to the desired targets. Figure 2 displays the simplified process of the BP algorithm.

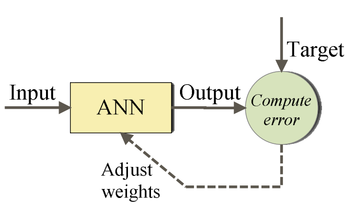


Figure 2. Block diagram of the BP algorithm.

III. Particle Swarm Optimization

Been introduced originally by Kennedy and Eberhart [7]-[11] in the field of studying social and cognitive behavior, the PSO is a population-based optimization approach, where one of the populations is called a swarm and each swarm consists of many particles. In the PSO, the trajectory of each particle in the search space is adjusted by dynamically altering their velocities. Each particle has a velocity vector and a position vector, which represents a possible solution. Then, the particles move rapidly around and search the solution space according to the moving velocities. Each of these particle positions is scored in order to obtain a fitness value, based on how to define the solution of the chosen application. The local best position (*Lbest*) of each particle and the global best position (*Gbest*) in the swarm are used to yield a new velocity for each particle, which is defined by

$$\begin{aligned}\vec{v}_i(k+1) = \omega\cdot\vec{v}_i(k) + \varphi_1\cdot\text{rand}()\cdot(L_{best} - \vec{x}_i(k)) \\ +\varphi_2\cdot\text{rand}()\cdot(G_{best} - \vec{x}_i(k)),\end{aligned} \tag{1}$$

where ω, φ_1, and φ_2 are called the coefficients of the inertia term, the cognitive term, and the society term, respectively, rand() is a function that returns random number, and $-\vec{v}_{\max} \le \vec{v}_i \le +\vec{v}_{\max}$. If the velocity violates this limit, then it is set to the closest boundary value. Changing the velocity enables each particle to search around its individual best position and global best position. Based on the updated velocities, each particle changes its position according to

$$\vec{x}_i(k+1) = \vec{x}_i(k) + \vec{v}_i(k+1). \tag{2}$$

Figure 3 presents the concept of the updated velocity by using Eqs. (1) and (2). In our proposed scheme, the PSO is utilized to learn optimal weights of the ANN in order to achieve better recognition efficiency in comparison to the BP algorithm.

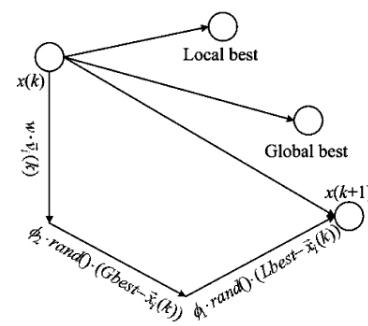


Figure 3. Diagram of the updated velocity in the PSO.

IV. Preprocess Tasks for Chord Recognition

Music is an actual part of life, and it can be any succession of single tones or simultaneous tones. When these tones blend harmoniously, a chord takes place involuntarily. Although such chords are an illustration of harmony, the chords are further used to accompany a piece of melody or be an arrangement of music. As the aim of this work is to implement chord recognition intelligently, the original music composition shall be extended with new materials or a fleshing-out of a compositional sketch, such as a lead sheet.

4.1 Development of Data for Training and Testing

For an efficient supervised training by applying ANNs, all the songs are segmented into phrases, while the duplicate ones should be eliminated. The phrase is the smallest musical section that expresses a complete idea, just like a sentence of a speech. In the great majority of cases, one phrase consists of four measures. To develop experimental data, these melodic phrases were recorded by playing the MIDI keyboard. Due to individual differences of human beings, the durations of each played notes are not always long enough. Duration and pitch of notes within a measure are essential factors for the arrangement of chords. Hence after recording the songs, all notes are quantized to fill up with full duration. The following subsections provide the relative detailed descriptions of the processes for MIDI data, cadences, and chord arrangement.

4.1.1. Processing MIDI Data

Music is an art that deals with Time. A definition of its metrical structure was given in Percy Goetschius [6], where the units of music metrical structure are not inches and the like, but divisions of time, i.e., the basis of which is the beat. The beats are grouped in measures of uniform durations, and every measure contains equal numbers of beats. In this research, we choose 84 children's ballads in a 4-4 measure. The time-signature 4-4 means quarter-note as beat and 4 beats per measure. Therefore, MIDI data is transformed into 16-digit patterns per measure, while one beat consists of 4-digit data. Because one scale includes 12 notes, the pitch information is transformed into numeric data between 1 and 12. Notes with the same pitch but different scale are treated as identical ones. The codes of notes are shown in Figure 4 and Figure 5 displays the formation of transformed MIDI data.

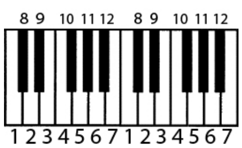


Figure 4. Codes of notes.

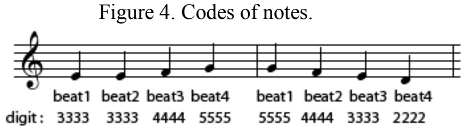


Figure 5. Formation of transformed MIDI data.

4.1.2. Cadence

As it was mentioned before, the performance of ANN training wasn't good enough, and therefore, one additional feature called cadences was added to the existing numeric data for each measure. Cadence in a melodic phrase is usually considered to be the important feature for chord recognition. For example, Percy Goetschius [6] referred that a cadence is the ending of a melodic phrase. Every interruption or "break" between all melodic phrases is a cadence. In other words, two typical cadences can be identified from the structural punctuation of melodic phrases. Conclusive cadence exhibits tonal closure at the end of a melodic phrase, and inconclusive cadence represents a temporary interruption for the succeeding tonal motion [13]. Cadences are usually employed in the 4^{th} measure within a phrase. According to the characteristics of cadences, we created six types of cadence feature number and assigned them to each measure. More precisely, cadence number "1" is assigned to the first measure in each musical phrase, while cadence number "4" is for the last measure in a conclusive phrase, and number "3" indicates the measure which is before measure number "4". In the same way, cadence number "6" is appointed to the last measure in an inconclusive phrase and number "5" is with the measure which is before measure number "6". The last cadence number "2" is used to represent the measures which are between "1 and 3" or "1 and 5". Detailed definitions of cadence feature numbers are summarized in Table 1, while Figure 6 presents an example of cadence numbers.

TABLE 1. DEFINITION OF CADENCE NUMBERS

Measures	Conclusive phrase	Inconclusive phrase
The first measure	1	1
Inner measures	2	2
Measure before cadence	3	5
Cadence measure	4	6

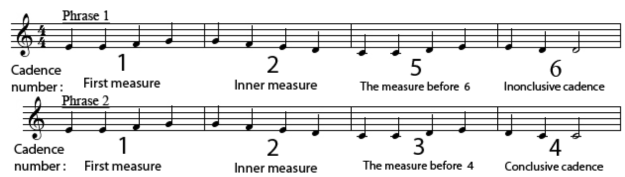


Figure 6. Cadence examples of actual cases.

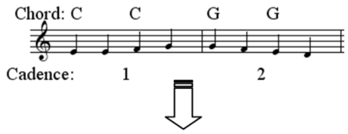
4.1.3. Chord Arrangement

As indicated in previous sections, the main objective of our proposed approach is to find out actual chord in accordance with the melody. Chord data is given as the target for training and testing datasets and further becomes the goal of actual outputs in the ANN model. In addition, chord data is referred to the music scores of children's ballads book. Because of proportion of beats within a measure, songs with 4-4 time signature are selected. Therefore, there are two chords in a measure, and one measure consists of two pieces of data. After all songs are transposed to C major, three types of chords used for experiments of this work are C, F, and G. Then the target chord data is represented with "C: 1 0 0", "F: 0 1 0", and "G: 0 0 1". Table 2 shows the example measure contains two chords, i.e., C and F.

TABLE 2. FORMATION OF CHORD DATA

<table>
<tr><th colspan="6">Measure</th></tr>
<tr><th colspan="3">Chord 1</th><th colspan="3">Chord 2</th></tr>
<tr><th>C</th><th>F</th><th>G</th><th>C</th><th>F</th><th>G</th></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
</table>

Finally, all the data of melody, cadence numbers and chords are combined and saved with TXT format to compose both the training and testing datasets. Example dataset within two measures is illustrated as in Figure 7.



	Data 1	Data 2
Measure 1	1 3333 3333	1 4444 5555
Chord(target)	100	100
Measure 2	2 5555 4444	2 3333 2222
Chord(target)	001	001

Figure 7. An example of two measures.

V. The Proposed Two-phase ANN Model for Chord Recognition

The simplified procedures of our proposed two-phase ANN model are presented as in Figure 8, while both the training and testing datasets are followed the extracting descriptions of preprocesses illustrated in the previous section. The two phases of our proposed approaches are the phase one for training and phase two for testing.

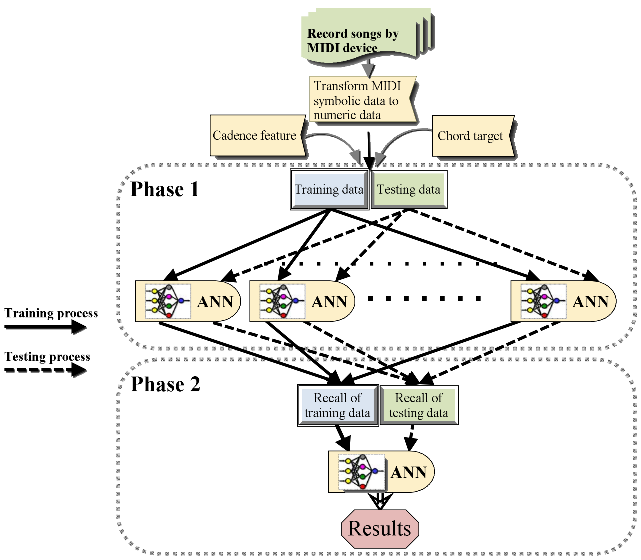


Figure 8. Block diagram of the proposed two-phase ANN model.

Since the aim of phase one is to calculate the averaged recognition rates of the training and testing datasets, there are five randomly initialized ANNs built, while the ANN with the optimal performance is selected to be the candidate for phase two.

In the processes of phase two, the five result datasets of the selected ANNs (consisting of recalls of training and testing datasets) are combined respectively and further compose the new training and testing datasets for the ANN training in phase two.

The diversity of these different initial weights is the leading cause of unsteady recognition results. Thus, we assume that these ANNs have different ability to recognize chords from each other. But in the proposed two-phase ANN model, we concern the characteristic of recognizing preference more than recognizing ability which is performed by each ANN in phase one. Say in other words, even if the ANN makes incorrect recognition to some particular patterns, the wrong results can be a significant recognizing feature for recognition in combining multiple ANNs. In next section, further details of our proposed two-phase model are illustrated by experimental simulations.

VI. Experimental Results

Both the training and testing datasets used in our experiments are generated from 84 children's ballads. The pattern numbers in the training and testing datasets are 1952 and 312, respectively, which consist of 1204 in chord C, 364 in chord F, and 696 in chord G.

6.1 Experimental Results of Phase One

In order to exhibit ANN learning efficiency, the experimental results of mean squared errors (MSEs) for the ANN-PSO (with n=30) learning is summarized as in Figure 9, where "n" is the number of hidden nodes in the ANN, "p" the amount of particles in the PSO, "non-cadence" means there is no cadence feature numbers contained in the training set, and "cadence" represents cadence feature number is included.

Table 3 presents the training recognition rates of the five ANNs with different parameter settings, i.e., with and without cadence features, and different number of particles, which are trained by the BP and PSO methods. From the averaged numerical results, it is easily observed that, by added the cadence features into consideration our proposed approach improves the recognition accuracies from 4.75% to 10.32% for ANN-PSO, and from 5.19% to 6.41% for ANN-BP. In addition, for both the cases with and without cadences, our proposed ANN-PSO outperforms ANN-BP in term of higher recognition rates, i.e., from 2.38% to 5.77% for the case with cadence, and from 1.09% to 5.64% for the non-cadence case.

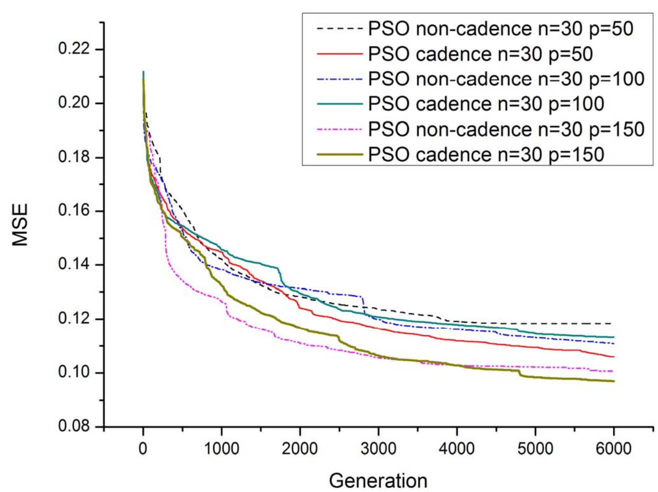


Figure 9. MSEs using ANN-PSO learning method.

Besides the numerical results indicated above, Figure 10 exhibits the comparisons of averaged recognition rates for the and "non-cadence" experiments trained by ANN-BP, and ANN-PSO with 50, 100, and 150 particles, respectively, while Figure 11 illustrates the case with cadence features.

From presented results in Table 3, it can be found that the best recognition occurs in ANN-PSO, with 100 particles and 40 hidden nodes, i.e., the averaged rate is 85.06%. At this stage, it is possible to speculate that the more PSO particles apply, the higher recognition rate will be achieved. However, our experimental result indicates that when the number of PSO particles increase to 150, the recognition rate becomes lower than the one of the 100-particle case. According to this, the optimal experimental parameters (i.e. the ANN-PSO with 100 particles) are chosen for applying to the next stage of the proposed two-phase ANN model.

TABLE 3. SIMULATION RESULTS FOR ANN-BP AND ANN-PSO

Parameters	Hidden Nodes	Experiments 1	2	3	4	5	Average (%)
BP non-cadence	20	71.15	71.47	73.40	68.27	66.99	70.26
	30	74.68	67.63	72.44	74.04	76.92	73.14
	40	73.40	73.72	74.68	73.08	69.87	72.95
BP cadence	20	76.28	75.00	77.24	77.88	75.64	76.41
	30	76.92	79.49	77.88	78.85	78.53	78.33
	40	78.85	77.88	79.17	81.09	79.81	79.36
PSO non-cadence Particles=50	20	76.60	75.00	74.68	76.28	72.44	75.00
	30	73.08	77.24	75.64	78.21	75.64	75.96
	40	75.96	76.92	74.68	69.55	76.92	74.81
PSO cadence Particles=50	20	79.49	81.09	82.05	75.32	81.41	79.87
	30	83.97	80.13	78.53	80.77	80.13	80.71
	40	83.97	82.05	83.97	83.65	83.33	83.40
PSO non-cadence Particles=100	20	74.68	78.53	75.32	73.72	77.24	75.90
	30	75.00	77.24	72.44	80.13	78.85	76.73
	40	71.15	75.96	74.68	77.24	74.68	74.74
PSO cadence Particles=100	**20**	**81.73**	**79.49**	**84.29**	**82.69**	**82.69**	**82.18**
	30	**82.37**	**85.26**	**83.33**	**84.29**	**84.29**	**83.91**
	40	**85.90**	**86.86**	**83.01**	**86.22**	**83.33**	**85.06**
PSO non-cadence Particles=150	20	74.04	71.15	78.53	74.04	78.21	75.19
	30	76.92	72.76	77.24	70.51	73.72	74.23
	40	76.60	71.79	72.12	73.72	76.28	74.10
PSO cadence Particles=150	20	81.41	78.85	84.62	78.53	80.77	80.83
	30	78.53	85.26	80.13	80.13	81.41	81.09
	40	83.65	81.09	82.69	83.65	78.85	81.99

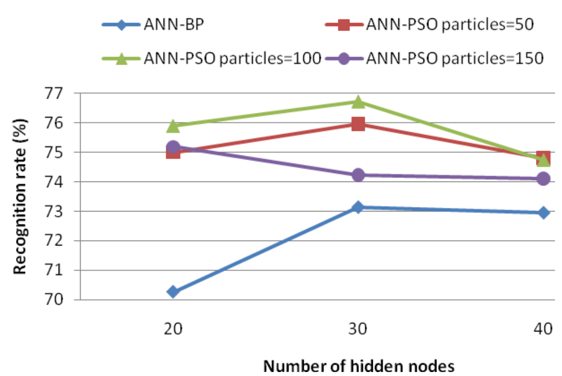


Figure 10. Experimental results with no cadence features.

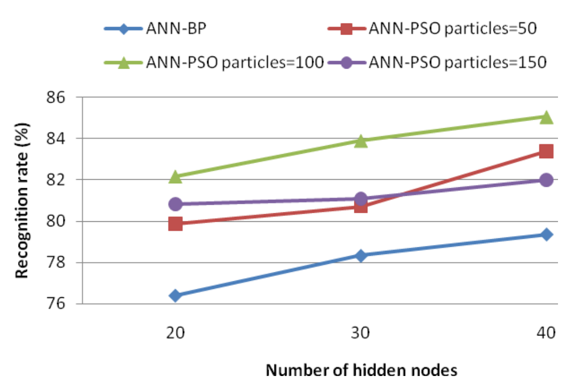


Figure 11. Experimental results with cadence features.

6.2 Experimental Results of Phase Two

As indicated previously the candidate ANN for our phase two is the ANN-PSO with 100 particles. By using 20, 30 and 40 hidden nodes these experiments were executed for five times in phase one, each of the five output datasets (composed by recalls of training and testing data) were combined to form the new training and testing datasets for the 2nd phase of our proposed model.

The parameters of the ANN in phase two are ANN-PSO, particles=100, nodes=40. The comparison results are tabulated in Table 4. Each experiment in phase two was also executed for five times to acquire the average recognition rate. The average results performed by two-phase ANN are better than single ANN performed in our early experiments, especially when n=20 and n=30. The results indicate that the proposed two-phase ANN model improved recognition accuracy significantly.

TABLE 4. RECOGNITION RATE OF OUR PROPOSED APPROACH

Method	Hidden Nodes	Experiments 1	2	3	4	5	Average (%)
ANN using PSO	20	81.73	79.49	84.29	82.69	82.69	82.18
	30	82.37	85.26	83.33	84.29	84.29	83.91
	40	85.90	86.86	83.01	86.22	83.33	85.06
The two-phase ANN using PSO	20	85.90	84.94	86.86	84.62	85.58	85.58
	30	84.94	84.94	86.54	84.62	85.58	85.32
	40	85.90	**88.14**	86.86	85.58	85.90	86.48

6.3 Discussions

From the presented simulation results above, it can be easily observed that the averaged improvement of our proposed approach for using 20, 30 and 40 hidden nodes are about 3.40%, 1.41% and 1.42%. By evaluating the numerous simulation results, there is strong evidence that using two-phase ANN combined with five ANNs can effectively avoid incorrect recognition results. In other words, due to this characteristic, our proposed two-phase ANN model can outperform any single ANN, in term of higher recognition accuracy.

In addition, we selected the 2nd set of results from the experiments of the two-phase ANN with 40 hidden nodes, since this set of simulations achieved the best recognition rate (88.14%) among all experiments (as shown in Table 4). Then we discovered the actual recognition results between the 124th to the 144th testing data, as illustrated in Figures 12-17. In the figures for ANN-1 to ANN-5, it can be observed that the incorrect recognition outputs distributed randomly in certain testing sets. When considering a two-phase ANN model to implement the fusion of ANN-1 to ANN-5, the corresponding results are to be more accurate. Furthermore, as depicted in Figure 17, when multiple ANNs were combined with the two-phase ANN model, the results were revised to the correct recognition. Therefore, it is obvious that the proposed two-phase ANN is capable of avoiding incorrect recognition resulting from the use of the traditional majority voting method or only applying one single ANN.

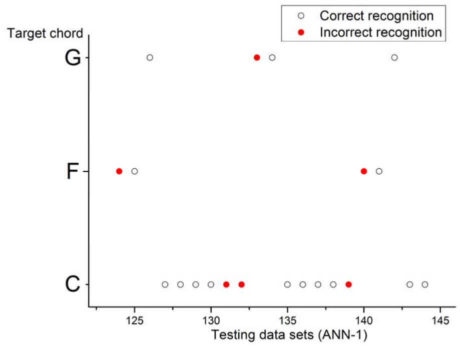


Figure 12. Distribution of chosen testing data points, recognized by ANN-1.

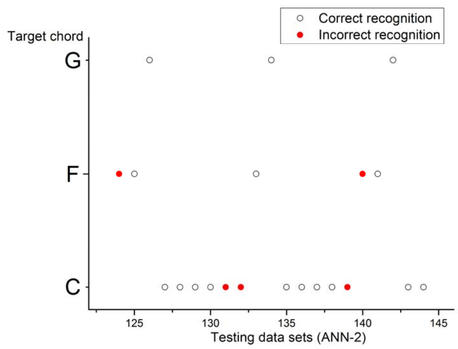


Figure 13. Distribution of chosen testing data points, recognized by ANN-2.

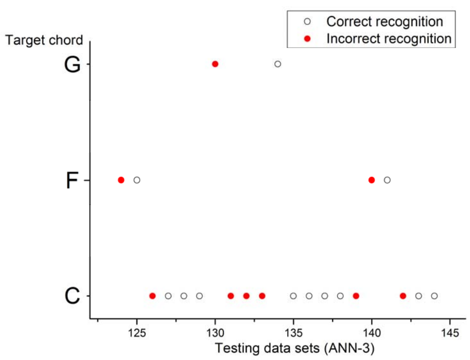


Figure 14. Distribution of chosen testing data points, recognized by ANN-3.

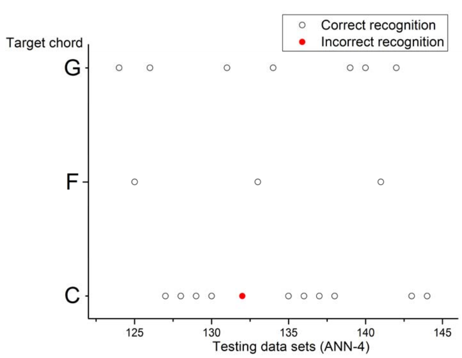


Figure 15. Distribution of chosen testing data points, recognized by ANN-4.

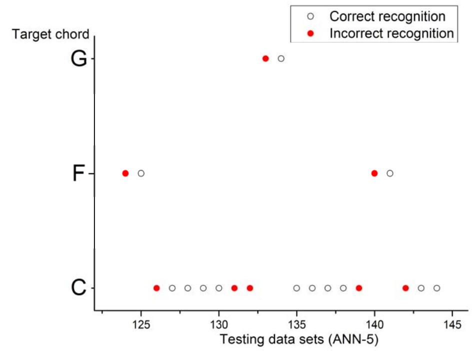


Figure 16. Distribution of chosen testing data points, recognized by ANN-5.

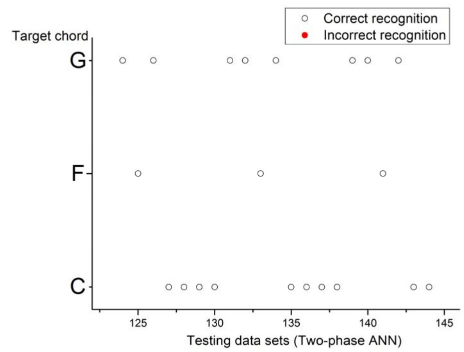


Figure 17. Distribution of chosen testing data points, recognized by Two-phase ANN.

VII. Conclusions and Future Works

It is known that music is the presentation of art which consists of infinite combinations of musical notes and time. In order to implement chord recognition intelligently, we tried to establish and derive the rules of numerous combinations from limited amounts of training datasets in this paper. Our proposed approach is to apply the two algorithms, i.e. the BP and PSO, to obtain the optimal weights of an ANN architecture. By comparing the experimental results, it is found that the ANN-PSO training has more promising results. On the other hand, we consider the incorporation of another important feature into training and testing datasets, which is the cadence number. Our experimental results show that by using cadence features the recognition rates improve significantly. In this paper, we also introduce a two-phase ANN model in order to obtain more stable experimental results and higher recognition accuracy. After examination of the simulation results of this two-phase model, it is obvious that the proposed two-phase ANN model implemented the fusion of multiple ANNs efficiently and combined multiple output results effectively to avoid incorrect recognition.

Among various kinds of music and chords, we chose regular and simple songs to compose the training and testing datasets in the first stage of our experiments. In future research works, more songs with different content and styles will be considered and developed. With the increased amounts of training and testing datasets, chords with more styles can be recognized and the generalization of our approach can be further verified.

REFERENCES

[1] I. Simon, D. Morris and S. Basu, "Exposing Parameters of a Trained Dynamic Model for Interactive Music Creation," *Association for the Advancement of Artificial Intelligence*, 2008.

[2] K. Lee and M. Slaney, "Automatic Chord Recognition from Audio Using a Supervised HMM Trained with Audio-from-Symbolic Data," in *Proc. Audio Music Comput. Multimedia Workshop* (*AMCMM*), pp.11-19, 2006.

[3] M. Geethanjali, A. Mary Raja Slochanal and R. Bhavani, "PSO trained ANN-based differential protection scheme for power transformers," *Neurocomputing*, vol. 71, pp. 904-918, 2008.

[4] D. Gang, D. Lehman and N. Wagner, "Tuning a Neural Network for Harmonizing Melodies in Real-Time," in *Proc. Int'l Computer Music Conference (ICMC)*, 1998.

[5] N. Chung, *My favorite ballads*, Mei-Yue Publishing Co., 1984.

[6] L. Chen, D.H. Cooley and J. Zhang, "Possibility based fuzzy neural networks and their application to image processing," *IEEE Trans. Systems Man and Cybernetics*, vol. 29, no. 1, pp. 119-126, 1999.

[7] C. T. Lin and C. S. G. Lee, *Neural Fuzzy Systems: A neural-fuzzy synergism to intelligent systems*. Englewood Cliffs, NJ: Prentice-Hall, May 1996.

[8] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int'l Conf. Neural Netw.*, 1995, pp. 1942-1948.

[9] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proc. 6th Int'l Symp.MicroMach. Hum. Sci.*, 1995, pp. 39-43.

[10] C. J. Lin, C. H. Chen, and C. T. Lin, "Hybrid of Cooperative Particle Swarm Optimization and Cultural Algorithm for Neural Fuzzy Network and Its Prediction Applications," *IEEE Trans. Systems, Man, and Cybernetics--Part C: Applications and Reviews*, vol. 38, no. 1, pp. 55-68, Jan. 2009.

[11] C. J. Lin, "An efficient immune-based symbiotic particle swarm optimization learning algorithm for TSK-type neuro-fuzzy networks design," *Fuzzy Sets and Systems*, vol. 159, pp. 2890-2909, 2008.

[12] P. Goetschius, *Lessons in Music Form*, Boston: Oliver Ditson, pp. 26-48, 1904.

[13] R. Gauldin, *Harmonic Practice in Tonal Music*, W. W. Norton & Company, pp. 36-40, 1997.