Blind Source Separation Research Based on the Feature Distance Using Evolutionary Algorithms

Yang Yang
School of Engineering, Bohai University, 121013, Jinzhou, China

Xiuqin Wang
Information Science and Technology School, Bohai University, 121013, Jinzhou, China

Di Zhang
Liaoning Datang International Jinzhou Power Generation Co., Ltd, 121001, Jinzhou, China

(Received 3 April 2012; revised: 28 August 2013; accepted: 14 January 2014)

Without any information on the mixing system, the blind source separation (BSS) technique efficiently separates mixed signals. The approach called evolutionary algorithms was used for the BSS problem in this paper. The fitness function based on the feature distance and kurtosis was proposed to measure the degree of the separated signals in this paper. Compared with the traditional algorithm in the BSS problem, the mathematical calculation and the physical significance of the separated signals are both taken into consideration in the proposed method. Therefore, the separated signals could have great correlation with the original individual signal and could be used in the additional signal processing step with good signal property. Experimental results on mixed spoken signals indicated that the established evolutionary algorithm of particle swarm optimization (PSO) and genetic algorithm (GA) could effectively solve the BSS problem from the signal feature distance and independence measurement. The study in this paper was implemented with MATLAB language.

1. INTRODUCTION

Without knowing the mixing processing and sources, blind source separation (BSS) deals with recovering a set of underlying sources from observations. The BSS problem is widely used in the fields of: image processing, acoustics signal separation, vibration signal separation, medical signal processing, biomedical data analysis, telecommunications, stock analysis and fault recognition.[1–3]

In the literature, the theory of BSS has been approached in several ways and various algorithms have been proposed. For example, the methods were originally introduced in the context of neural network (NN) modelling, independent component analysis (ICA), principle component analysis (PCA), singular value decomposition (SVD), high order statistical cumulants and others. The most important and simplest of the methods mentioned above is ICA which has the goal of finding a suitable representation of non-Gaussian sources with all the most independent components as possible. Lots of ICA algorithms for BSS problems are proposed, including the minimization (or maximization) of a contrast function (for example Mutual Information and non-Gaussianity). ICA works with different algorithms, including FastICA algorithm, JADE (Joint Angle and Delay Estimation) algorithm, extended Infomax algorithm, and mean field approach ICA. The ICA method differs from other similar methods in that the components are both statistically independent and non-Gaussian. BSS is used for recovering unobserved signals from a known set of mixtures. Therefore, ICA and BSS are equivalent when the mixtures are assumed to be linear up until possible permutations and invertible scalings.[2–7]

In the past, the NN model was the popular architecture for separation, but its performance depends strongly on the initiation of weight. In a previous study, the authors used the genetic algorithm (GA) for optimizing the weights of the NN system in order to enhance global convergence.[8] In another study, a support vector machine (SVM) methodology is applied to ICA in the search for the separating matrix.[5]

According to a previous paper, through finding optimum and accurate coefficients of the separating matrix, the evolutionary algorithms can be the best solution for solving BSS problems. In this approach, the new population can be created where independence among its components is maximized if a suitable fitness function is used. There are two types of contrast functions of BSS: information theory and high order statistics. In this paper, the authors used two evolutionary algorithms, GA and PSO, for BSS, and the novel fitness function is based on the mutual information and high order statistics.[2]

In another paper the authors present a novel GA-ICA method which converges to the optimum.[9] The new method uses GA to find the separating matrices, which are based onthe contrast function to minimize a cumulant. In reference 10 the authors used the kurtosis of the mixed signal to the target function, by modifying PSO to replace the steepest gradient descent method. In reference 11 the learning rate of the BSS method is selected adaptively by using PSO. In reference 12 the authors introduce the evolution speed and the aggregation degree to update the dynamic inertia weight in PSO. In reference 13, for blind deconvolution and the deblurring of images, the method is based on a non-Gaussian measure of ICA along with the GA for optimization in the frequency domain.

In this paper, the BSS approach for linear mixed signals is studied to get the coefficients of the separating matrix by using evolutionary algorithms (PSO and GA). The operation of these algorithms principally depends on the fitness function by using the kurtosis and the feature distance, which will be defined later. We first constructed two mixed signals using two spoken word signals. The objective is to separate the signals from the mixed ones and this is a typical BSS problem. Then we used evolutionary algorithms to separate the mixed matrix. The simulation results showed that a good result can be obtained by using the feature distance combined with kurtosis as the fitness function. Kurtosis is a simple and necessary criterion for estimation dependency among signals. The proposed method not only uses the mathematical way to find the optimal matrix, but it also takes into consideration the signals' own characteristics, as can be seen in the feature distance definition. When the feature distance and the independence of estimated signals are at a maximum, the two signals are separated well. Other simulation results also showed that the proposed method is valid and can be used in the similar field.

2. BLIND SOURCE SEPARATION AND EVOLUTIONARY ALGORITHMS

2.1. BSS Problem Description

A series of observed signals is given, and BSS aims at recovering the underlying sources by using the assumption of their mutual independence. BSS can be classified as linear or nonlinear based on the type of mixing of the sources.

The BSS model considered in this paper is a linear simultaneous mixture in Eq. (1).

$$\mathbf{x} = \mathbf{A}\mathbf{s}; \tag{1}$$

where $\mathbf{x} = [x_1, x_2, \ldots, x_m]^T \in R$ is a vector containing measured signals x_i, $\mathbf{s} = [s_1, s_2, \ldots, s_n]^T \in R^n$ is a vector containing original sources $(m \geq n)$, and $\mathbf{A} \in R^{m \times n}$ is an unknown mixing matrix with full column rank.[1] The linear model can also be expressed as in Eq. (2):

$$x_j(t) = a_{j1}s_1 + a_{j2}s_2 + \cdots + a_{jn}s_n \quad \forall j = 1 \ldots n. \tag{2}$$

Assume that the number of sources n is equal to that of mixtures m. For simplicity, the discussion here is restricted to the case of $m = n = 2$. In the experiment we will construct two mixed signals using two original spoken signals. Certain assumptions about sources are also needed in the BSS problem. The most general ones are:[1]

1. Sources are mutually independent;
2. Sources are non-Gaussian or one Gaussian signal at most;
3. The mixing matrix is a full unknown column rank.

With the above assumptions, the BSS result has two inherent ambiguities:[1]

1. The order of the estimated sources cannot be decided;
2. Original variances (energies) of sources are unknown.

Therefore, all the sources are generally assumed to have unit variances.[1]

The matrix W (the separating matrix) whose output can be an estimate of the sources $s(t)$ is given in Eq. (3):

$$\mathbf{y} = \mathbf{W}\mathbf{x} \tag{3}$$

In ICA, a solution that maximizes the non-Gaussianity of the recovered signals is needed. Therefore, some ways to measure the non-Gaussianity are also required including negentropy and kurtosis.

Negentropy is used as a measure of distance to normality in information theory. The entropy of a discrete signal is equal to the negative sum of the products of the probability of each event and the log of those probabilities. Kurtosis is a classical method of measuring non-Gaussianity which is equal to the fourth moment of the data if the data is pre-processed with unit variance. In an intuitive sense, kurtosis is used to measure the "spikiness" of a distribution or the size of the tails. It is extremely simple to calculate but sensitive to outliers in the data set at the same time.[13] Mathematically kurtosis, is defined in Eq. (4).:[13]

$$Kurt(y) = E\{y^4\} - 3\left(E\{y^2\}\right)^2. \tag{4}$$

If y has unit variance, we can obtain $Kurt(y) = E\{y^4\} - 3$. If x_1 and x_2 are random variables, $Kurt(x_1 + x_2) = Kurt(x_1) + Kurt(x_2)$ and $Kurt(ax) = a^4 Kurt(x)$ are satisfied.

2.2. Particle Swarm Optimization (PSO)

The PSO method was developed by Eberhart and Kennedy in 1995.[14] It simulates social behaviour to a promising position in order to achieve precise objectives in a multi-dimensional space. The PSO method has been applied in a wide variety of highly complicated optimizations in real-world problems. Like other evolutionary algorithms, PSO performs searches using a population (called a swarm) of individuals (called particles) that are updated from iteration to iteration. Each particle changes its search direction based on two factors to discover the optimal solution. The first one is its own best previous experience and the other one is the best experience of all other members.[14–16]

The basic process of the PSO algorithm is initialization, fitness, update, construction, and termination. The process of PSO is finished if the termination condition is satisfied. The details are given as follows:[16]

1. Generate initial particles randomly;
2. Measure the fitness of each particle in the population;
3. Compute the velocity of each particle;
4. Move to the next position for each particle;

5. Stop the algorithm if the termination criterion is satisfied; otherwise, return to Step 2.

The position vector and the velocity vector of i th particle in an m-dimensional search space can be represented as $x_i (i = 1, 2, \ldots, N)$ and $v_i (i = 1, 2, \ldots, N)$ respectively; N represents the number of particles.

In the PSO algorithm, the new velocities of other particles are updated by Eqs. (5) and (6).

$$v_i(t+1) = \omega v_i(t) + c_1 r_1 (p_i(t) - x_i(t)) + c_2 r_2 (p_g(t) - x_i(t)); \quad (5)$$

$$x_i(t+1) = x_i(t) + v_i(t+1); \quad (6)$$

where v_i is the velocity of the i th particle of the swarm, x_i is the position in the search space. p_i is the best position of the i th particle, p_g is the global best particle, ω is the inertia weight of velocity, c_1 and c_2 are the acceleration coefficients, and r_1 and r_2 are two different, uniformly distributed random numbers in the range of $[0, 1]$. The potential of the solution is measured by the fitness function in our paper. More details about the PSO algorithm can be seen in the reference section.[8,17–19]

2.3. Genetic Algorithm (GA)

The GA is one of the most popular stochastic optimization techniques nowadays. The GA method is inspired by the natural genetics and biological evolutionary process. Three basic operators are used to manipulate the genetic composition of a population: reproduction, crossover and mutation. The GA evaluates a population and generates a new one iteratively with each successive population (generation).[6]

The goal is to solve the optimization problem. Here, the chromosome is written as an array with an n-dimensional optimization problem and can be seen in Eq. (7).[20]

$$chromosome = [p_1, p_2, p_3, \ldots, p_n]. \quad (7)$$

Each chromosome has a cost found by evaluating the fitness function f at the variables $p_1, p_2, p_3, \ldots, p_n$.

$$f(chromosome) = f(p_1, p_2, p_3, \ldots, p_n) \quad (8)$$

The GA algorithm is characterized as follows:[6]

1. Encodes solutions to a problem in the form of a chromosome;
2. Initializes the population for the chromosomes procedure;
3. Evaluates fitness function;
4. Manipulates the composition of the population using genetic operators;
5. Provides the initial settings of the population size and probabilities employed by the genetic operators.

3. EXPERIMENTS

In the experiment, two spoken word signals ($kiss1$ and $love1$) were used as the individual signals.[21] Suppose that $y1$ is the name of the $kiss1$ signal and $y2$ is the name of the $love1$ signal. We know that two spoken word signals do not have the same length most of time. Therefore, we add several zero values at the end of the short signal to make their length same. Then two mix signals were constructed, which are $mix1 = 0.3 * y1 + 0.5 * y2$ and $mix2 = 0.4 * y1 + 0.3 * y2$. The mixed signals are the weighted sums of the original spoken signals; the weights depend upon the distances between the source signals and the microphones. Here the mixing matrix was chosen randomly.The unknown matrix is square, and the mixing can be characterized by a linear scenario. The objective is to separate the individual signals from the mixed ones, and this is a typical BSS problem. The recovered signals are called $ys1$ and $ys2$ in this paper.

3.1. Fitness Evaluation-Kurtosis

Kurtosis is used to measure the degree of the non-Gaussian property of the signals. The common evolutionary algorithm in the BSS problem is based on the kurtosis calculation. Preprocessing of the BSS data is needed before using kurtosis as the fitness function which contains two steps: centring and whitening.[2]

The fitness function is defined as follows in Eq. (9):

$$F(ys) = |kurt(ys1)| + |kurt(ys2)|$$
$$fitness = -F(ys) \quad (9)$$

The kurtosis of a distribution in MATLAB 7.0 is defined in Eq. (10):

$$k = \frac{E(x-\mu)^4}{\sigma^4} \quad (10)$$

where μ is the mean of x, σ is the standard deviation of x, and $E(t)$ represents the expected value of the quantity t.

3.2. Fitness Evaluation- Feature Distance

As signal has its own characteristic and features vectors that can help distinguish speech signals. There is more than one way to choose the feature vectors. For the spoken signals, DFT coefficients were used as features.[2] Other popular alternatives include the parameters from an AR modelling of the speech segment and the ceptstral coefficients (the inverse DFT of the logarithm of the magnitude of the DFT coefficients). In the experiment, the AR modelling method was selected as the feature. When the feature distance of two recovered signals is at its maximum, the two signals are separated well by the algorithm.

The feature distance is defined in Eq. (11):

$$F(ys) = \sum_{j=1}^{m} (|f(ys_1)| - |f(ys_2)|); \quad (11)$$

where f is the feature function, and m is the number of the feature vectors. The fitness function is defined in Eq. (12).

$$fitness = -F(ys) \quad (12)$$

3.3. Fitness Evaluation– kurtosis and feature distance

With the advantage of kurtosis and feature distance as the fitness function, the separated signal can have both physical significance and the independence property. Therefore, we proposed a new fitness function combined with kurtosis and feature distance to improve the algorithm; it is defined in Eq. (13):

$$F(ys) = |k(ys_1)| + |k(ys_2)| + \sum_{j=1}^{m} (|f(ys_1)| - |f(ys_2)|)$$
$$fitness = -F(ys); \tag{13}$$

where f is the feature function and m is the number of the feature vectors.

3.4. Simulation Result

We individually used Particle Swarm Optimization (PSO) and the Generic Algorithm (GA) to settle the BSS problem.

In the PSO method, the particles were used in the separating matrix. Here we chose the learning factor synchronization of the PSO algorithm to separate the mixed spoken signals in the experiment. Three fitness functions were used to test the algorithm. The waveforms of the source signals and the recovered signals can be seen from Fig. 1 to Fig. 3.
Kurtosis(y1)=26.9992
Kurtosis(y2)=30.4246
Kurtosis(ys1)=30.4824
Kurtosis(ys2)= 27.0729
Fitness function value= -57.4238 (expected value)
Fitness function value= -57.5553 (experimental value)
Fitness function value = -6.6461 (expected value)
Fitness function value = -6.6728 (experimental value)
Fitness function value = -64.0699(expected value)
Fitness function value = -64.1792 (experimental value)

In the GA method, the program was written with the Genetic Algorithm Tool in MATLAB. The population size was 40, the variable number which used in the separating matrix was 4 and the other parameters were by default. Three fitness functions were also used to test the algorithm we proposed in the paper.
Kurtosis(y1)= 26.9992
Kurtosis(y2)= 30.4246
Kurtosis(ys1)= 27.0767
Kurtosis(ys2)= 30.4782
Fitness function value = -57.4238 (expected value)
Fitness function value = -57.5548 (experimental value)
Fitness function value = -6.6461(expected value)
Fitness function value = -6.6439 (experimental value)
Fitness function value = -64.0699(expected value)
Fitness function value = -64.1792 (experimental value)

The recovered signals $ys1$ and $ys2$ were obtained by using the optimal separating matrix, whose figures can be seen in Figs. 4 to 6. Compared with the classical ICA algorithm 2 (FastICA, Hyvarinen's fixed-point algorithm). The signals can be seen in Fig. 7.

In order to evaluate and compare the performance of BSS, the correlation analysis and the source to distortion ratio (SDR) were used to verify the similarity between the source signals yi and separated signals ysi with N samples. SDR is defined as in Eq. (14).

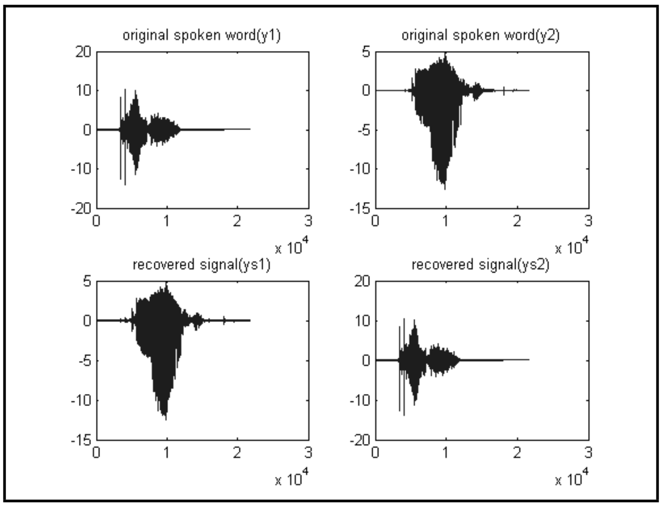

Figure 1. PSO-kurtosis

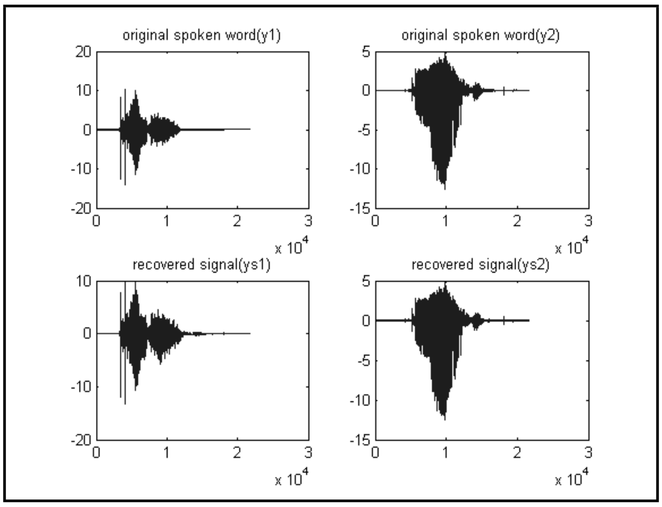

Figure 2. PSO-feature distance

$$SDR(yi, ysi) = 10log\left(\sum_{t=1}^{N} \frac{[yi(t)]^2}{\sum_{t=1}^{N} [ysi(t) - yi(t)]^2}\right); \quad (14)$$

where the larger the SDR is, the better the effect of separated signals is.

The experimental results of the PSO, GA, and FastICA method were given for comparison in Table 1.

From Table 1 we can see that, the proposed algorithm with the fitness of kurtosis and the feature distance has good results that are similar with FastICA. The key point in the performance of the evolutionary algorithm is the definition of the fitness function. The separated method in the paper uses not only the mathematical way to find the optimal matrix, but also takes into consideration the signals' own characteristics. After doing the similarsimulation, the simulation result of the proposed method is still effective.

Table 1. Result analysis between the signals recovered and the source signals

Signals	Source Signals			
Recovered	$y1$	$y2$	SDR	algorithm
ys1	0.0259	0.9998	SDR(y1,ys2)= 31.7390	PSO
ys2	0.9997	0.0209	SDR(y2,ys1)= 33.6021	f: kurtosis
ys1	0.9592	0.3271	SDR(y1,ys2)= 10.8874	PSO
ys2	0.0447	1.0000	SDR(y2,ys1)= 53.7254	f: feature distance
ys1	0.0369	1.0000	SDR(y1,ys1)= 28.6550	PSO
ys2	0.9993	0.0099	SDR(y2,ys2)= 40.1285	f: feature distance& kurtosis
ys1	-0.9996	-0.0185	SDR(y1,ys2)= 30.9869	GA
ys2	0.0282	0.9998	SDR(y2,ys1)= 34.6346	f: kurtosis
ys1	0.0576	0.9999	SDR(y1,ys1)= 9.0815	GA
ys2	-0.9382	-0.3895	SDR(y2,ys2)= 39.2657	f: feature distance
ys1	0.0364	0.9999	SDR(y1,ys1)= 28.7721	GA
ys2	0.9993	0.0103	SDR(y2,ys2)= 39.7033	f: feature distance& kurtosis
ys1	0.0094	0.9993	SDR(y1,ys2)= 40.5799	FastICA
ys2	1.0000	0.0374	SDR(y2,ys1)= 28.5384	

Note: f means fitness function.

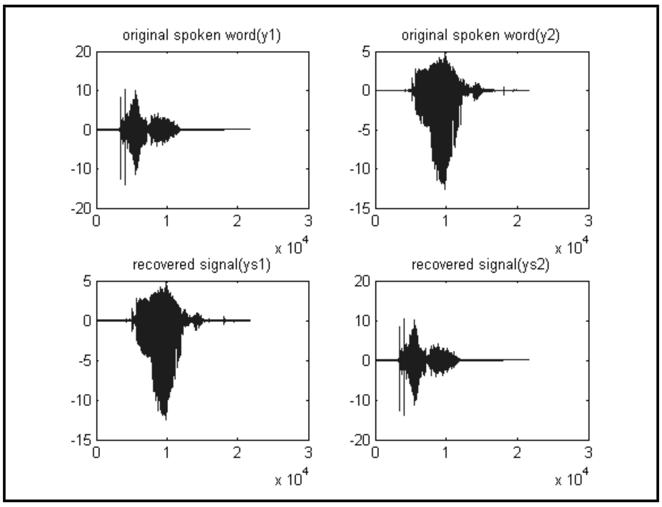

Figure 3. PSO-kurtosis and feature distance

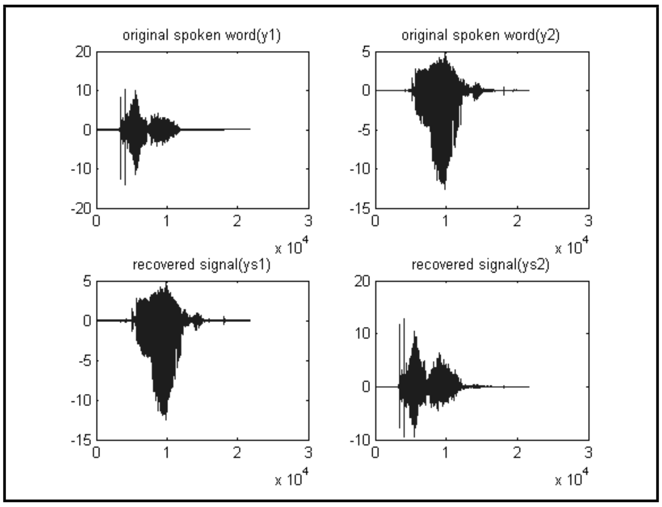

Figure 5. GA-feature distance

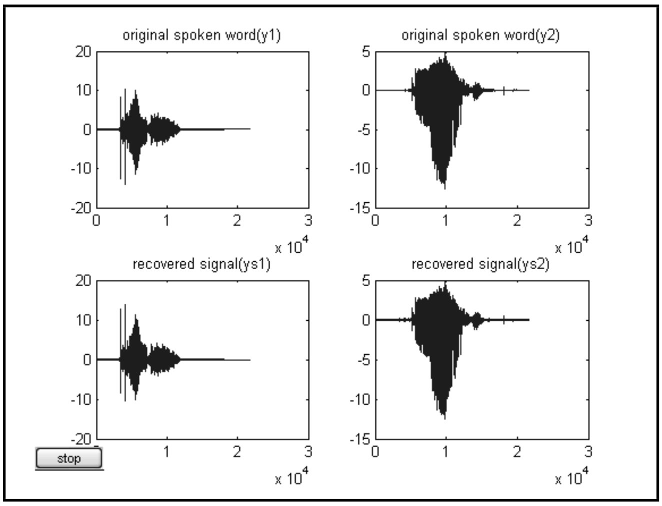

Figure 4. GA-kurtosis

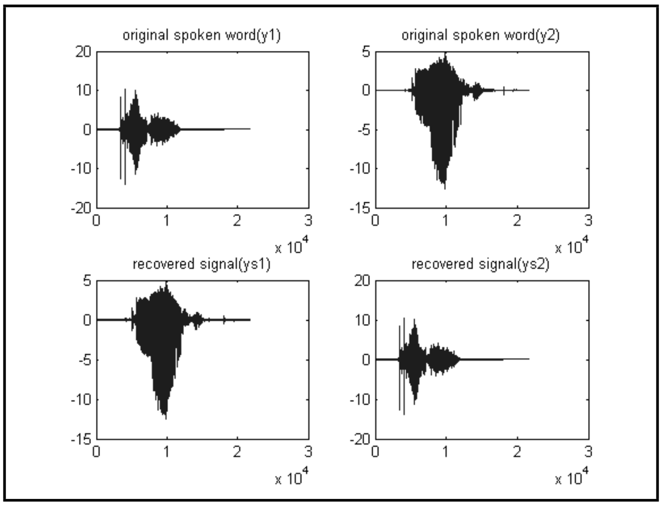

Figure 6. GA-kurtosis and feature distance

4. CONCLUSION

BSS is a good method for dealing with mixed signals. Individual source signals can be obtained if the separating assumptions are satisfied. By introducing the evolutionary method with the feature distance and kurtosis as the fitness function in the experiment, the separated signals can have both physical significance and the independence property. It can be widely used in the BSS problem, evolution algorithm, signal processing, and similar research. Our further study will be the evolutionary algorithm on the nonlinear mixing models in the BSS problem.

ACKNOWLEDGEMENT

The authors thank the support of Education Department of Liaoning Province, China. The project number is L2010006.

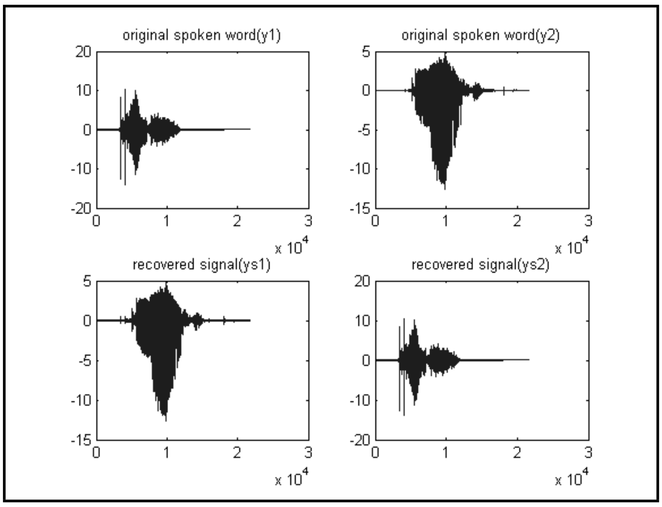

Figure 7. Fast-ICA

REFERENCES

[1] Zhou, W. L. and Chelidze, D. Blind source separation based vibration mode identification, *Mechanical Systems and Signal Processing*, **21**(8), 3072–3087, (2007).

[2] Mavaddaty, S. and Ebrahimzadeh, A. Research of Blind Signals Separation with Genetic Algorithm and Particle Swarm Optimization Based on Mutual Information, *Journal of Computer Engineering*, **1**, 77–88, (2009).

[3] Yang, F. S. and Hong, B. The theory and application of Individual Component Analaysis, *Tsinghua University Press*, (2006).

[4] Hyvärinen, A., Karhunen, J. and Oja, E. Independent component analysis, John Wiley & Sons, Inc, (2001).

[5] Puntonet, C. G., Górriz, J. M. and Moiśes, S. et al. Theoretical Method for Solving BSS-ICA Using SVM, *Fifith International Conference*, ICA 2004, Granada, Spain, (2004).

[6] Rojas, F., Puntonet, C. G. and Álvarez, M. R. et al, Evolutionary Algorithm Using Mutual Information for Independent Component Analysis, *7th International Work-Conference on Artificial and Natural Neural Networks*, Maó, Menorca, Spain, (2003).

[7] Vosough, M. Using mean field approach independent component analysis to fatty acid characterization with overlapped GCMS signals, *Analytica Chimica Acta*, **598**, 219–226, (2007).

[8] Liu, C. C., Sun, T. Y. and Hsieh, S. T. et al. A Hybrid Blind Signal Separation Algorithm: Particle Swarm Optimization on Feed-Forward Neural Network, *International Conference on Neural Information Processing*, Hong Kong, China, (2006).

[9] Górriz, J. M., Puntonet, C. G. and Morales, J. D. et al. Simulated Annealing Based-GA Using Injective Contrast Functions for BSS, *Proceedings of the 5th International Conference on Computational Science*, Atlanta, GA, USA, (2005).

[10] Zhang, W. A., Liu, J. H. Blind Source Separation Based on Particle Swarm Optimization, *Journal of Taiyuan University of Technology*, **37**(2), 169–172, (2006).

[11] Sathyapriya, N., Balaji, N. and Lokesh, T. L. et al. Blind Source Separation Using Improved PSO For Adaptive Adjustment Of Learning Rate, *Conference Proceedings RTCSP'09*, (2009).

[12] Li, M., Li, W. J., and Wang, Y. et al. Blind Source Separation Based on Improved Particle Swarm Optimization, *Artificial Intelligence and Computational Intelligence*, Shanghai, China, (2009).

[13] Yin, H.J., Hussain, I. ICA and Genetic Algorithms for Blind Signal and Image Deconvolution and Deblurring, *Intelligent Data Engineering and Automated Learning 2006*, Burgos, Spain, (2006).

[14] Kennedy, J. and Eberhart, R. Particle swarm optimization, *Proc IEEE International Conference on Neural Networks*, Nagoya, (1995).

[15] Paul, R. C., Asokan, P. and Prabhakar, V. I. A solution to the facility layout problem having passages and inner structure walls using particle swarm optimization, *Int J Adv Manuf Technol*, **29**, 766–771, (2006).

[16] Lin , S.W., Chen, S.C. and Wu, W. J. et al. Parameter determination and feature selection for back-propagation network, *Knowl Inf Syst*, **21**, 249–266, (2009).

[17] Gao, Y., Li, Z. H. and Zheng, H. et al. Extended Particle Swarm Optimiser with Adaptive Acceleration Coefficients and Its Application in Nonlinear Blind Source Separation, *International Conference on Neural Information Processing*, Hong Kong, China, (2006).

[18] Fernandez-Martinez, J.L. and Garcia-Gonzalo, E. Stochastic Stability Analysis of the Linear Continuous and Discrete PSO Models, *IEEE Transactions on Evolutionary Computation*, **15**(3), 405–423, (2011).

[19] Song, K., Ding, M. L. and Wang, Q. et al. Blind Source Separation in Post-nonlinear Mixtures Using Natural Gradient Descent and Particle Swarm Optimization Algorithm, *4th International Symposium on Neural Networks*, Nanjing, China, (2007).

[20] Haupt, R. L. and Haupt, S. E. Practical Genetic Algorithms, *John Wiley & Sons*, Inc., Hoboken, New Jersey, (2004). Second edition.

[21] Theodoridis, S. and Koutroumbas, K. Pattern Recognition, *China Machine Press*, Beijing, (2006). Third edition.