BLIND SIGNAL SEPARATION/DECONVOLUTION USING RECURRENT NEURAL NETWORKS

Yan Li and David Powers

School of Informatics and Engineering
The Flinders University of South Australia, Australia
{yan.li, david.powers}@flinders.edu.au

ABSTRACT

A new iterative learning algorithm based on 2D system theory using recurrent neural networks (RNNs) is presented for blind signal separation in this paper. The characteristics of convolutive (real) signals are matched by the structure of RNNs. The feedback paths in a RNN can memorise the past signals (echoes) so that better separation can be achieved. The cross-correlations of the outputs of the RNN are used as separation criterion. The experimental results for artificially mixed data and real multi-channel recordings demonstrate the performances of the algorithm.

1. INTRODUCTION

Blind signal/source separation (BSS) aims to extract the underlying source signals from a set of their unknown mixtures. It's growing applications lie in various fields, such as telecommunication systems, sonar and radar systems, audio and acoustics, image enhancement and biomedical signal processing. Much success has been achieved in the use of neural networks for blind signal separation. These methods include the Hebbian learning algorithm [1], robust adaptive algorithm [2], nonlinear principal component analysis [3], information maximisation [4] and simple neuron models for independent component analysis [5].

Although the potential of the powerful mapping and representation capabilities of recurrent network architectures is generally recognised by the neural network research community. RNNs have not been widely used for blind signal separation and deconvolution, possibly due to the relative ineffectiveness of the simple gradient descent training algorithm. Actually, RNNs can perform better separation for real signals than simple neural networks as the dynamic structure of an RNN matches the characteristics of convolutive signals better. The feedback paths in an RNN can memorise the past signals (echoes). Amari et al [6] demonstrated their RNN needs a much smaller operational range for the synaptic weights compared with the feedforward network used.

This paper presents an on-line approach using RNNs for blind signal separation. The new unsupervised learning algorithm based on 2D system theory is proposed. We use output decorrelation as the signal separation criterion to minimise the statistical correlation between outputs of the network. Many different approaches have been presented by numerous researchers using output decorrelation [7], [8]. Gerven and Compernolle [7] stated that decorrelating the output signals at different time lags is sufficient provided that the normalised auto-correlation functions of the source signals are sufficiently distinct.

The rest of the paper is organised as follows: The architecture of RNNs and the algorithm is presented in the next section. Simulations for different signal separation are obtained in section 3. Finally, the paper is concluded in section 4.

2. LEARNING ALGORITHM

2.1 Signal separation by output decorrelation

Blind signal separation is the process of extracting unknown source signals from sensor measurements, which are unknown combinations of the sources. For instantaneous mixing, all received signals are the linear superposition of the sources, namely, the outputs of microphones. Suppose that source signals are denoted by a vector $\underline{S}(t) = [S_1(t), S_2(t), \ldots, S_n(t)]^T$, $t = 0, 1, 2\ldots$. And the observed signals are denoted by $\underline{X}(t) = [X_1(t), X_2(t), \ldots, X_n(t)]^T$, $t=0, 1, 2\ldots$. We have general equations:

$$\underline{X}(t) = \underline{A}(t)\cdot \underline{S}(t) \qquad (1)$$
$$\underline{Y}(t) = \underline{W}(t)\cdot\underline{X}(t) \qquad (2)$$

When $\underline{W}(t) = \underline{A}^{-1}(t) \quad \Rightarrow \quad \underline{Y}(t) = \underline{S}(t)$

here, $\underline{A}(t)$ is the unknown mixing matrix and $\underline{Y}(t) = (Y_1(t), Y_2(t), \ldots, Y_n(t))^T$ is the separated signals. The task is to recover the original sources by finding a matrix $\underline{W}(t)$ ($\underline{W}(t)$ may be time-varying), which theoretically is a permutation and rescaling of the inverse of the unknown matrix $\underline{A}(t)$, so that $\underline{Y}(t)$ is as close as possible to $\underline{S}(t)$. In convolutive BSS, a source is corrupted by time-delayed versions of itself and other source signals. In this case, the Equations (1) and (2) hold in the frequency domain. Taking the z-transform of Equations (1) and (2), we have

$$\underline{X}(z) = \underline{A}(z)\underline{S}(z)$$
$$\underline{Y}(z) = \underline{W}(z)\,\underline{X}(z)$$

for convolutive BSS.

Output decorrelation is a simple approach to the signal separation problem in which the source signals S_i are assumed to be zero-mean and independent. This approach exploits the temporal correlations of the signals. The objective is to produce uncorrelated outputs y_i by minimising an arbitrary number of cross-correlation lags. The cross-correlation function of two signals, $y_i(t)$ and $y_j(t)$ is

$$Ry_i y_j(l) = \frac{1}{B-l}\sum_{k=1}^{B-l} y_i(k)\,y_j(k+l) \qquad (3)$$
$$i \neq j,\ l = 0, 1, 2, 3, \ldots, L$$

here B is the number of the sampling points of the data taken. Equation (3) is used as the cost function in this paper to derive a new iterative learning algorithm based on 2D system theory for an RNN with time varying weights. The weights of the RNN can be estimated in a real-time fashion, and the cross-correlation cost function can reach a small level as close as zero. The algorithm can dynamically search for the best lag(s) from a wide range for an optimum separation result.

2.2 Recurrent Neural Networks

In an RNN, basic processing units are fully connected so that there are both feedforward and feedback paths. Nodes in an RNN are generally classified into three categories (instead of layers): input, output, and hidden nodes. In this paper, we use processing nodes to represent all the output nodes and hidden nodes. Processing nodes receive output signals from all nodes including themselves. There are two sets of synaptic connections in RNNs. The first set of connections indicated by weight matrix $\underline{W_2}=\{w_{ij}\}$ link the input and the processing nodes. $w_{ij}(t)$ denotes the strength of the connection from the j^{th} input node to the i^{th} processing node, at time t. The second set of connections indicated by weight matrix $\underline{W_1}=\{w_{ij}*\}$ form the feedback paths. Similarly, $w_{ij}*(t)$ denotes the strength of the connection from the j^{th} processing node to the i^{th} processing node, at time t.

Let $\{y_i(t)\}$ denote the outputs of the processing nodes and $\{u_j(t)\}$ denote the external inputs. We have

$$S_k(t) = \sum_{l\in I} w_{kl}(t)u_l(t) + \sum_{l\in U} w_{kl}(t)y_l(t) \qquad (4)$$

and $$y_k(t+1) = f_k[S_k(t)] \qquad (5)$$

where $S_k(t)$ is the activation of the k^{th} processing node at time t and $f[]$ is usually a sigmoidal function. Obviously, Equations (4) and (5) can also be represented in the following matrix form:

$$\underline{y}(t+1) = f[\underline{W_1}(t)\,\underline{y}(t) + \underline{W_2}(t)\,\underline{u}(t)] \qquad (6)$$

with initial value $\underline{y}(0)= y_0$. (7)

where

$$\underline{y}(t) = [y_1(t), y_2(t),..., y_n(t)]^T \in R^n,$$

$$\underline{u}(t) = [u_1(t), u_2(t),..., u_m(t)]^T \in R^m,$$

$$\underline{W_1} \in R^n x R^n, \quad \underline{W_2} \in R^n x R^m$$

$\underline{f}(\cdot)$ is a vector of a nonlinear activation function,

An algorithm based on a two-dimensional expression, which will be introduced in the next section updates the connection weights to drive the cross-correlations to zero.

2.3 The learning algorithm based on 2D theory

In the algorithm for the RNN, there is a dynamic process described by the evolution of the outputs in terms of time t. For each time t, the learning algorithm executes several times, modifying the network weights at each iteration k. After a number of learning iterations, the network at time t should obtain appropriate weights to drive the network outputs to be independent at that time-step. Then there is also another dynamic process described by the change of variable k to reflect the learning iterations. Therefore, during the learning process, each variable of an RNN depends upon two independent variables: discretized time t and the learning iteration k. For example, $y(t, k)$ and $\underline{W_1}(t,k)$ represent the network outputs and weights in the tth time-step of the kth learning iteration. According to the 2D expression [9], the RNN model (6) can be rewritten as

$$\underline{y}(t+1,k) = f[\underline{W_1}(t,k)\underline{y}(t,k) + \underline{W_2}(t,k)\underline{u}(t,k)] \qquad (8)$$

This is a 2D dynamic system that clearly describes the 2D dynamic behaviour of iterative learning of the RNN.

The derivation of the learning algorithm is based on the algorithm proposed by Chow and Fang [10]. Chow and Fang [10] used an error equation, which expresses the differences between the target outputs and actual outputs to derive a supervised learning algorithm for dynamic control systems. In their paper, they proved that the learning algorithm can drive the cost function as small as zero after one learning iteration ($k=1$). But this algorithm does not work in our case as the signal separation is a form of unsupervised learning. For our purpose, the cross-correlations of the outputs of the RNN are employed as signal separation criterion. The goal is to drive the cost function to be zero. It is an unsupervised learning algorithm.

The iterative learning rule can be expressed as follows:

$$\Delta\underline{W}(t+1) = \underline{C}^{-1}(t)Cd[\underline{x}(t)^T\underline{x}(t)]^{-1}\underline{x}(t)^T \qquad (9)$$

where Cd is cross-correlations of outputs shown in equation (3), and

$$\underline{C}^{-1}(t) = [diag(f'(\underline{\xi_1}), f'(\underline{\xi_2}),...., f'(\underline{\xi_n}))]^{-1},$$

$$\underline{\xi} = (\underline{\xi}_1, \underline{\xi}_2,..., \underline{\xi}_n)^T = \underline{W}(t)\underline{x}(t),$$

$$\underline{x}(t) = \begin{pmatrix} \underline{y}(t) \\ \underline{u}(t) \end{pmatrix}, \quad \underline{y}(t+1) = f[\underline{W}(t)\underline{x}(t)],$$

$$\underline{W}(t+1) = \underline{W}(t) + \Delta\underline{W}(t)$$

For each lag $l = 0, 1, 2, 3 .... L$, the algorithm minimises the corresponding cross-correlation recursively. The choice of the parameter l many depends on an individual application. Therefore, we can obtain the outputs of the network:

$$\underline{y}(t+1) = f[\underline{W}(t)\underline{x}(t)]$$

The algorithm is efficient as the iterative learning algorithm is able to drive the cross-correlations to zero. Comparing other neural algorithms in BSS/ICA with (9), the real-time learning algorithm has a dynamic learning rate of $\underline{C}^{-1}(t)[\underline{x}(t)^T\underline{x}(t)]^{-1}$. The feedback paths in the RNN can memorise the past signals (echoes), which matches the characteristics of convolutive signals. In the next section, the simulation results demonstrate the algorithm for RNNs is a very promising algorithm for BSS.

3. SIMULATION RESULTS

The simulation results for several signals using the proposed approach are given in this section. The observed signals are simultaneously mixed signals, convolutively mixed signals, and real environment recordings. For the simplicity of presentation, only the results of the two-input/two-output signals are demonstrated. The algorithm can be easily extended to more than two inputs and outputs cases.

The proposed algorithm in this paper is an on-line algorithm that separates the signals sample by sample. Furthermore, the approach used in this study is non-linear. It is not proper way to use the weights as the main objective measure as many researchers have done (such as in [3], [7]) due to the weights that are finally available are the weights that separate the last sample of the input signals. The experiments are assessed qualitatively by listening to and viewing the waveforms.

Experiment 1 Artificially mixed signals

One recording of male speech $S_1(t)$ and one recording of female speech $S_2(t)$ were recorded separately. They were artificially mixed together. The mixed signals $X_1(t)$ and $X_2(t)$ are in the form of

$$\begin{bmatrix} X_1(t) \\ X_2(t) \end{bmatrix} = \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & -0.5 \end{bmatrix} \begin{bmatrix} S_1(t) \\ S_2(t) \end{bmatrix}$$

An optimal result is obtained when the block size B is greater than 200. And the lag is 5 in this case. Good results are still available when any more value(s) of lag are included, but the training time would be longer since more recursions are carried out. The separated signals are comparable with the original signals when listened to. For this simple case, the waveforms of the original, mixed and separated signals are not included in this paper due to the limited spaces.

Experiment 2 Covolutively mixed signals

The same two recordings used in Experiment 1 were used here but with longer data segments. They were mixed as below:

$$X_1(t) = S_1(t) + 0.3S_2(t-1)$$
$$X_2(t) = S_2(t) + 0.3S_1(t-1)$$

The original and convolutively mixed signals are shown in figure 1 and 2. The block size of this experiment is 1000, and the lags used for the results in figure 3 are 1 and 6.

Experiment 3 Real Recordings

The observed signals were taken from Dr Te-Won Lee's home page at the Salk Institute on the website http://www.cnl.salk.edu/~tewon/. One signal is a recording of the digits from one to ten spoken in English. The second microphone signal is the digits spoken in Spanish at the same time. The proposed algorithm is applied to the signals. Figures 4 and 5 show the real signals and the separated results (we only present half of the signals here for clarity). It is hard to compare the results with Lee's results in a quantitative way due to the different methodologies, but comparable results can be identified when the signals are listened to.

The block size for cross-correlations was 500, and the correlation lags of 6 and 11 were used in this experiment for the optimal results. The training procedure has been repeated 3 times for the results in figure 5. The learning factor has been adapted from 1 to 0.001 in order to search widely for the correct weights, and then converge closely to them.

4. CONCLUSION

A new iterative learning algorithm based on 2D system theory using an RNN is presented for blind signal separation in this paper. The characteristics of convolutive (real) signals are matched by the structure of an RNN. The feedback paths in an RNN can memorise the past signals (echoes) so that better separation can be achieved. The cross-correlations of the outputs of the RNN are used as separation criterion. The experimental results for artificial data and for the real recordings demonstrate the algorithm performs well.

5. REFERENCES

[1]. C. Jutten and J. Herault, Blind separation of sources, Part I: An adaptive algorithm based on neurominic architecture, Signal Processing, Vol. 24, pp.1-10, 1991.

[2]. A. Rummert Cichocki, R. Unbehauen, Robust Neural Networks with on-line learning for blind identification and blind separation of sources, IEEE Trans Circuits and Systems I: Fundamentals theory and Applications, Vol. 43, No.11, pp. 894-906, 1996.

[3]. E. Oja and J. Karhunen, Signal separation by non-linear Hebbian learning, In M. Palaniswami, et al (editors), Computational Intelligence – A Dynamic System Perspective, IEEE Press, New York, NY, pp.83-97, 1995.

[4]. Anthony J Bell and Terrence J. Sejnowski, An information-Maximisation approach to blind separation and blind deconvolution, Neural Computation, Vol.7, No.6, pp.1004-1034, 1995.

[5]. A. Hyvärinen and E. Oja. Independent Component Analysis by General Non-linear Hebbian-like Learning Rules. Signal Processing, Vol. 64, No. 3, pp. 301-313, 1998.

[6]. S. Amari, A. Cichochi and H. H Yang, Recurrent Neural Networks For Blind Separation of Sources, 1995 International Symposium on Nonlinear Theory and its Applications, NOLTA'95, Las Vegas, December, pp.37-42, 1995.

[7] D. Van Compernolle and S. Van Gerven, On use of decorrelation in scalar signal separation, In IEEE, ICASSP, Vol. 3, pp.57-60, 1994.

[8]. Dominic C. B Chan, Blind Signal Processing, PhD thesis, Signal Processing and Communications Laboratory Department of Engineering, University of Cambridge, UK, 1997.

[9]. T. Kaczorek and J. Klamka, Minimum energy control of 2D linear systems with variable coefficients, Int. J. Control, Vol. 44, No. 3, pp. 645-650, 1986.

[10]. Tommy W S Chow and Yong Fang, A Recurrent Neural Network-Based Real-Time Learning Control Strategy Applying to Nonlinear Systems with Unknown Dynamics, IEEE Transactions Industrial Electronics, Vol.15, No.1, pp.151-161, February, 1998.

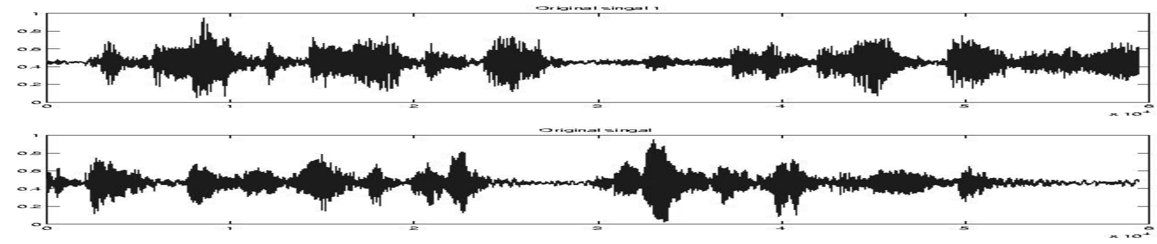

Figure 1 The original signals for the Experiment 2

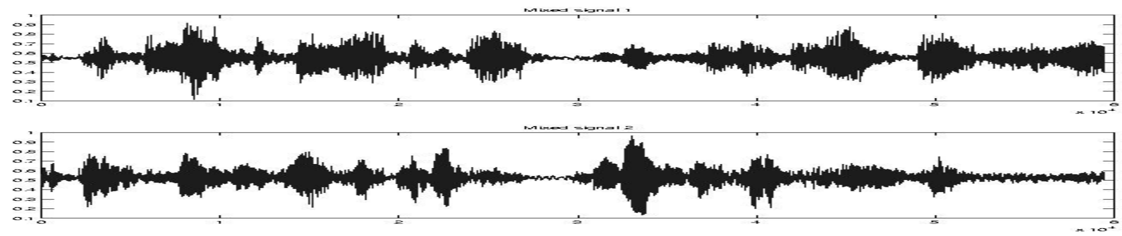

Figure 2 The mixed signals for the Experiment 2

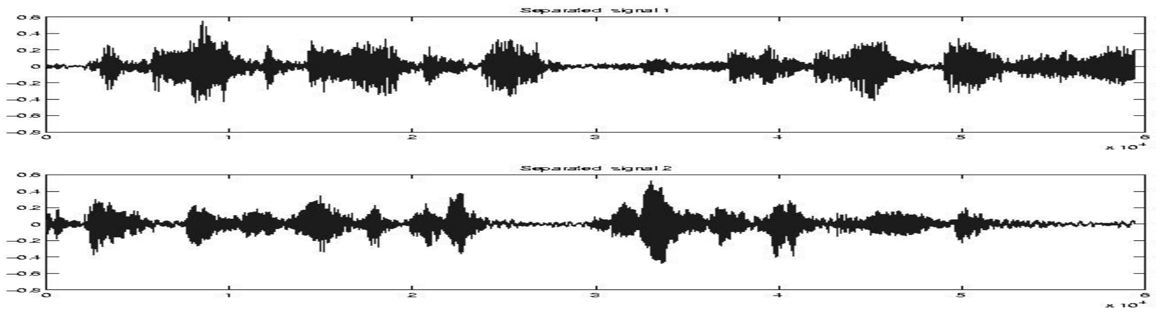

Figure 3 The separated signals for the Experiment 2

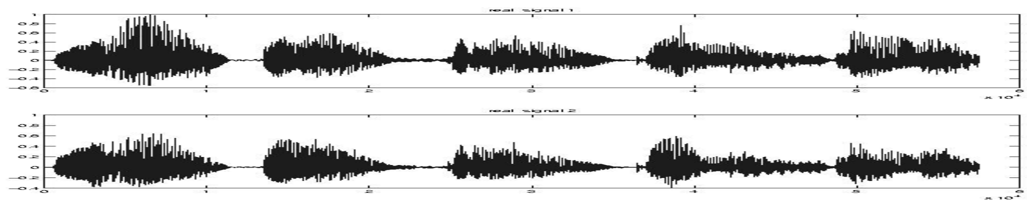

Figure 4 The real recordings for the Experiment 3

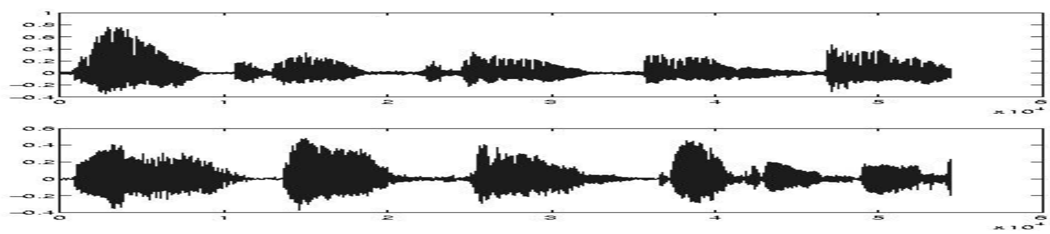

Figure 5 The separated signals for the Experiment 3