KALMAN FILTERING ALGORITHM FOR BLIND SOURCE SEPARATION

Qi Lv and Xian-Da Zhang[*]

Tsinghua University
Department of Automation
Beijing 100084, China
zxd-dau@tsinghua.edu.cn

Ying Jia

Intel China Research Center
Beijing, China
ying.jia@intel.com

ABSTRACT

This paper presents a Kalman filtering algorithm based on nonlinear principal component analysis (PCA) with prewhitening for blind source separation (BSS), and compare the new algorithm with other algorithms. The simulations show that the Kalman filtering algorithm has the faster convergence rate and the much better tracking capability, as compared with the existing natural gradient algorithm for independent component analysis (ICA), the RLS algorithm and the natural gradient based RLS-type one for nonlinear PCA for BSS.

1. INTRODUCTION

In recent years, blind source separation (BSS) has drawn lots of attention in signal processing community and neural networks community. BSS can be implemented via either independent component analysis (ICA) or nonlinear principal component analysis (PCA).

All the existing ICA algorithms are of least-mean-square (LMS) type. In contrast, the nonlinear PCA has the LMS-type algorithm [1] and the RLS-type one [2]. Recently, Zhu and Zhang [3] have developed a RLS-type algorithm based on natural gradient for nonlinear PCA in BSS.

Several comparison studies (see e.g. [2], [3]) have shown that the RLS type algorithms for BSS perform better than the LMS type algorithms for BSS in convergence rate and tracking capability. On the other hand, it is well-known ([4]) that the RLS is a special case of the Kalman filter, and the Kalman filter is well known for its tracking ability. A question comes out naturally that whether one can develop the Kalman filtering algorithm for BSS. The aim of this paper is to provide a solution to this problem.

This paper is organized as follows: Section 2 describes the nonlinear PCA with prewhitening for BSS; Section 3 presents the novel Kalman filtering algorithm; and Section 4 shows the simulation results; finally, the conclusion is given in Section 5.

*This work was supported by the major programm of the National Natural Science Foundation of China under contract 60390541 and the Intel Sponsored Research Foundation.

2. BLIND SOURCE SEPARATION

In instantaneous BSS, the signal model is usually expressed as

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

where $\mathbf{x}_n = [x_1(n), \cdots, x_m(n)]^T$ is the mixture vector, $\mathbf{s}_n = [s_1(n), \cdots, s_p(n)]^T$ is the source signal vector, and $\mathbf{A} \in R^{m\times p}(m \geq p)$ is a mixing matrix. Here, we assume that the number p of source signals is known.

The following three assumptions are usually made [5]:

1. $\mathbf{A}$ is a constant matrix with full column rank.
2. The source signals $s_i(n)$ $(i = 1, \cdots, p)$ must be mutually independent or in practice as independent as possible.
3. Each source signal $s_i(n)$ is a stationary stochastic process with zero mean and unity variance. Only one of the source signals is allowed to have a Gaussian marginal distribution.

The objective of on-line BSS is to update adaptively the $p\times m$ separating matrix $\mathbf{B}_n$, such that the $p\times 1$ output vector

$$\mathbf{y}_n = \mathbf{B}_n\mathbf{x}_n \tag{2}$$

is a copy of the unknown source vector $\mathbf{s}_n$.

The most key task in BSS is the learning of the separating matrix $\mathbf{B}_n$. Roughly speaking, the currently existing algorithms for learning $\mathbf{B}_n$ can be divided into two main groups: the first group is to find $\mathbf{B}_n$ directly, while the second group uses prewhitening before learning. Let $\mathbf{U}_n$ be the $p \times m$ prewhitening matrix such that $\mathbf{v}_n = \mathbf{U}_n\mathbf{x}_n$ is the $p\times 1$ normalized white noise vector with the zero mean vector and the unity covariance matrix $E\{\mathbf{v}_n\mathbf{v}_n^T\} = \mathbf{I}$. Then, the separating matrix is given by $\mathbf{B}_n = \mathbf{W}_n^T\mathbf{U}_n$, and the output vector of the BSS system can be expressed by

$$\mathbf{y}_n = \mathbf{W}_n^T\mathbf{U}_n\mathbf{x}_n = \mathbf{W}_n^T\mathbf{v}_n. \tag{3}$$

The adaptive BSS consists of updating the $p \times m$ prewhitening matrix $\mathbf{U}_n$ and the $p \times p$ weight matrix $\mathbf{W}_n$, respectively.

By applying the relative gradient, Cardoso and Laheld [5] presented a LMS-type prewhitening algorithm, given by

$$\mathbf{U}_n = \mathbf{U}_{n-1} + \eta_n(\mathbf{I} - \mathbf{v}_n\mathbf{v}_n^T)\mathbf{U}_{n-1} \quad (4)$$

where η_n is a learning rate, and $\mathbf{v}_n = \mathbf{U}_{n-1}\mathbf{x}_n$ denotes the prewhitened output vector.

After prewhitening, the BSS task becomes somewhat easier, since the components of the prewhitened vector $\mathbf{v}_n$ are already uncorrelated statistically, and the statistical uncorrelation is a necessary prerequisite of statistical independence. Generally speaking, separation algorithms using prewhitening converge faster and have better stability properties [6].

3. NOVEL KALMAN FILTERING ALGORITHM

In this section, we consider the learning algorithm for the $p \times p$ weight matrix $\mathbf{W}_n$ in nonlinear PCA with prewhitening. A typical nonlinear PCA criterion is given by [7]

$$J(\mathbf{W}) = E\{\| \mathbf{v}_n - \mathbf{W}_n\mathbf{g}(\mathbf{W}_n^T\mathbf{v}_n) \|^2\} \quad (5)$$

where $\mathbf{g}(\mathbf{y}_n) = [g(y_1(n)), \cdots, g(y_p(n))]^T$, $g(\cdot)$ is some nonlinear function, and $E\{\cdot\}$ is the expectation operation. In the following, we establish the state space equation for nonlinear PCA with prewhitening.

3.1. Kalman Filtering Algorithm

A dynamitic system is defined (see [4]) by a process or state equation

$$\mathbf{x}_{n+1} = \mathbf{F}_{n+1,n}\mathbf{x}_n + \mathbf{v}_{1,n} \quad (6)$$

and a measurement equation

$$\mathbf{y}_n = \mathbf{C}_n\mathbf{x}_n + \mathbf{v}_{2,n} \quad (7)$$

where $\mathbf{x}_n$ and $\mathbf{y}_n$ are the state vector and observation vector at time n, respectively; $\mathbf{F}_{n+1,n}$ is a known state transition matrix relating the states of the system at times $n+1$ and n, $\mathbf{C}_n$ is a known measurement matrix, and $\mathbf{v}_{1,n}$ is called the process noise, whereas $\mathbf{v}_{2,n}$ represents a measurement noise.

To develop an adaptive BSS algorithm based on Kalman filtering, we need to devise a novel linear first-order state space model for the BSS problem. And the key in Kalman filter is to establish the state space equation for problem.

In BSS problem, we view the weight matrix $\mathbf{W}^T$ as the state matrix rather than the state vector. A fact in the stationary linear mixing case is that the optimum weight matrix $\mathbf{W}_{opt,n}^T$ is time-invariant. Hence, the $p \times p$ optimum weight matrix obeys the following state equation:

$$\mathbf{W}_{opt,n+1}^T = \mathbf{W}_{opt,n}^T. \quad (8)$$

From (5), we define $\mathbf{e}_n = \mathbf{v}_n - \mathbf{W}_{opt,n}\mathbf{g}(\mathbf{W}_{opt,n}^T\mathbf{v}_n)$ as the error vector. To get the measurement equation similar to (7), we denote $\mathbf{y}_n = \mathbf{W}_{opt,n}^T\mathbf{v}_n$ and rewrite the error vector in the row vector form as follows:

$$\mathbf{v}_n^T = \mathbf{g}^T(\mathbf{y}_n)\mathbf{W}_{opt,n}^T + \mathbf{e}_n^T. \quad (9)$$

The state space equation for BSS consists of the process equation (8) and the measurement equation (9). As compared with the standard state space equation consisting of the process equation (6) and the measurement equation (7), the state space equation for BSS has the following characteristics:

1. The state vector becomes the state matrix $\mathbf{W}_{opt,n}^T$, the state transition matrix $\mathbf{F}_{n+1,n}$ is the unity matrix, and the process noise equals zero.
2. The observation vector becomes the row vector $\mathbf{v}_n^T$, the measurement matrix $\mathbf{C}_n$ is simplified to the row vector $\mathbf{g}^T(\mathbf{y}_n)$, and the measurement noise vector $\mathbf{e}_n^T$ is a row vector as well.

From the above corresponding relationships, we can formulate the discrete Kalman filtering algorithm for nonlinear PCA as follows:

$$\begin{aligned}
\mathbf{z}_n &= \mathbf{g}(\mathbf{W}_{n-1}^T\mathbf{v}_n) = \mathbf{g}(\mathbf{y}_n), \\
\mathbf{h}_n &= \mathbf{K}_{n,n-1}\mathbf{z}_n, \\
\mathbf{m}_n &= \mathbf{h}_n/(\mathbf{z}_n^T\mathbf{h}_n + Q_n), \\
\mathbf{K}_{n+1,n} &= \mathbf{K}_{n,n-1} - \mathbf{m}_n\mathbf{h}_n^T, \\
\mathbf{W}_n^T &= \mathbf{W}_{n-1}^T + \mathbf{m}_n(\mathbf{v}_n^T - \mathbf{z}_n^T\mathbf{W}_{n-1}^T) \quad (10)
\end{aligned}$$

where $\mathbf{K}_{n,n-1} = E\{(\mathbf{W}_n^T - \hat{\mathbf{W}}_n^T)(\mathbf{W}_n^T - \hat{\mathbf{W}}_n^T)^T\}$ is the predicted state-error correlation matrix, and $Q_n = ||\mathbf{e}_n||^2 = ||\mathbf{v}_n - \mathbf{W}_{n-1}\mathbf{z}_n||^2$.

We remark that when the linear mixing process is slowly time-varying, the process equation becomes $\mathbf{W}_{n+1} \approx \mathbf{W}_n$, and thus the above Kalman filtering algorithm is available as well in slowly time-varying linear mixing situations.

3.2. Comparison with Other Algorithms

In the following, we make a comparison of our Kalman filtering algorithm with other existing ones for BSS.

A method combining Kalman filter and natural gradient algorithm is proposed in [8], but it is restricted to the separation of mixtures of binary distributed sources.

On the other hand, the existing nonlinear PCA algorithms can be divided into the two classes: the LMS-type nonlinear PCA subspace rule [1], [6] and the RLS-type nonlinear PCA algorithm [2]. The nonlinear PCA subspace rule is given by [9] [2]

$$\mathbf{W}_n = \mathbf{W}_{n-1} + \mu[\mathbf{v}_n - \mathbf{W}_{n-1}\mathbf{g}(\mathbf{y}_n)]\mathbf{g}(\mathbf{y}_n^T). \quad (11)$$

Using a nonlinear form of projection approximation subspace tracking (PAST) algorithm of Yang [10], the RLS-type nonlinear PCA algorithm is given by [2]

$$\begin{aligned}\mathbf{z}_n &= \mathbf{g}(\mathbf{W}_{n-1}^T\mathbf{v}_n) = \mathbf{g}(\mathbf{y}_n),\\ \mathbf{h}_n &= \mathbf{P}_{n-1}\mathbf{z}_n,\\ \mathbf{m}_n &= \mathbf{h}_n/(\mathbf{z}_n^T\mathbf{h}_n+\lambda),\\ \mathbf{P}_n &= \frac{1}{\lambda}\mathrm{Tri}(\mathbf{P}_{n-1}-\mathbf{m}_n\mathbf{h}_n^T),\\ \mathbf{W}_n &= \mathbf{W}_{n-1}+(\mathbf{v}_n-\mathbf{W}_{n-1}^T\mathbf{z}_n)\mathbf{m}_n^T \end{aligned}\quad (12)$$

where $0 < \lambda \leq 1$ is a forgetting factor close to unit, and $\mathrm{Tri}(\cdot)$ means that only the upper triangular part of argument is computed, and its transpose is copied to the lower triangular part, making the matrix $\mathbf{P}_n$ symmetric.

The following are main comparisons of the four nonlinear PCA algorithms:

1. Algorithm (11) is a LMS-type, both (12) and the natural gradient based nonlinear PCA in [3] belong to RLS-type, while (10) is a Kalman filtering-type algorithm.

2. The updation of the separating matrix $\mathbf{W}_n$ is controlled by the error term $\mathbf{v}_n - \mathbf{W}_{n-1}\mathbf{g}(\mathbf{y}_n)$ in the three algorithms, but the controlled factor is the nonlinear function $\mathbf{z}_n = \mathbf{g}(\mathbf{y}_n)$ in (11), the gain vector $\mathbf{P}_{n-1}\mathbf{z}_n/(\mathbf{z}_n^T\mathbf{P}_{n-1}\mathbf{z}_n+\lambda)$ in (12), and the Kalman gain vector $\mathbf{K}_{n,n-1}\mathbf{z}_n/(\mathbf{z}_n^T\mathbf{K}_{n,n-1}\mathbf{z}_n+Q_n)$ in (10), respectively.

4. SIMULATION RESULTS

In simulations, we assume that the number of source signal is known ($m = p$), and use the natural gradient algorithm in [11], the RLS-type one in [2], the RLS based on natural gradient (RLS-NG) one in [3] and this paper's Kalman filtering one at the same time.

To obtain faster convergence and better steady-state accuracy, the natural gradient algorithm uses the time-varying learning rate $\lambda_n = \eta_n \times 10^{-4}$, where

$$\eta_n = \begin{cases} 50, & 1 \leq n \leq 1500 \\ 50e^{5(1500-n)}, & n > 1500 \end{cases} \quad (13)$$

while the RLS and RLS-NG algorithms uses the same time-varying forgetting factor given by

$$\beta_n = \begin{cases} 0.95 + (n-1)\times 10^{-4}, & 1 \leq n \leq 500; \\ 1, & n > 500. \end{cases} \quad (14)$$

We prewhiten the data for all the algorithms except for the natural gradient algorithm. The nonlinear function $g(t) = \tanh(t)$ is used for natural gradient algorithm, while the nonlinear function $g(t) = t - \tanh(t)$ (see [12]) is used for the other algorithms.

The performance index (PI) is given by [3]

$$\mathbf{PI} = \sum_{i=1}^{m}\left(\sum_{j=1}^{m}\frac{|c_{ij}|}{\max_k |c_{ik}|}-1\right) + \sum_{j=1}^{n}\left(\sum_{i=1}^{n}\frac{|c_{ij}|}{\max_k |c_{kj}|}-1\right) \quad (15)$$

where c_{ij} is the elements of the combined mixing-separating matrix $\mathbf{C} = \mathbf{BA}$.

In the first group of simulations, we use the four speech source signals shown Fig. 1, which are taken from [13], and the mixing matrix $\mathbf{A}$ is generated randomly each time. The PI curves averaged over 1000 independent runs are drawn in Fig. 2.

In the second group of simulations, the mixing matrix is a time-varying matrix given by

$$\mathbf{A}_t = \begin{bmatrix} \cos 2\pi f t & \sin 2\pi f t \\ -\sin 2\pi f t & \cos 2\pi f t \end{bmatrix}$$

where $f = 1$Hz, while the two source signals are the above two signals in Fig. 1. The PI curves are shown in Fig. 3.

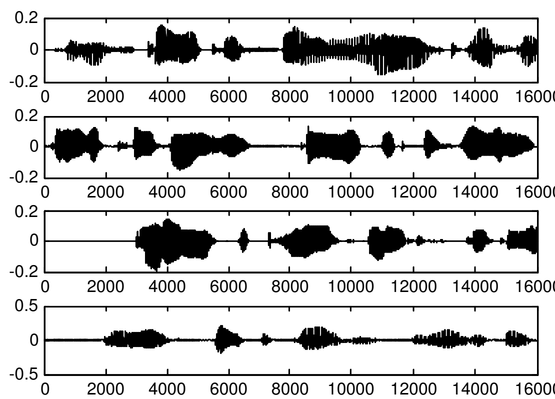

Fig. 1. The four speech source signals.

5. CONCLUSION

We have developed a new Kalman filtering algorithm for nonlinear PCA in BSS, and compared it with three existing methods. Simulations using the real speech source signals have shown that the Kalman filtering algorithm has the faster convergence rate and the better tracking capability, as compared with the EASI algorithm, the RLS-type one and the RLS-type one based on natural gradient.

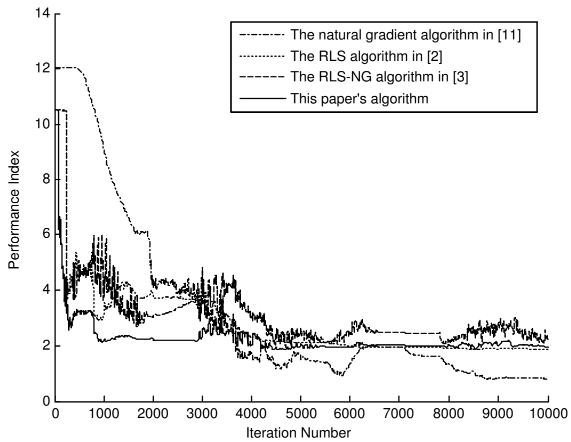

Fig. 2. The PI curves averaged over 1000 independent runs.

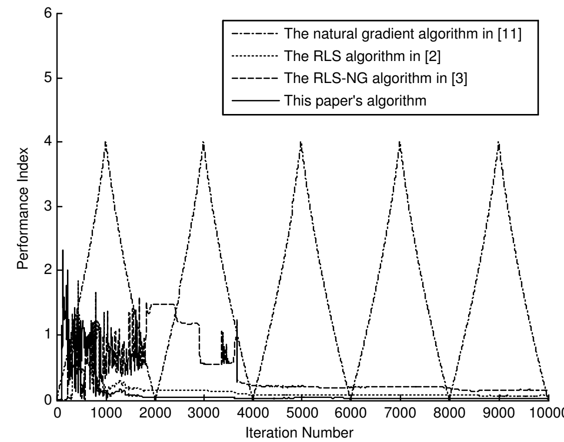

Fig. 3. The PI curves for time vary mixing matrix experiment.

6. REFERENCES

[1] E. Oja, "The nonlinear PCA learning rule in independent component analysis," *Neurocomputing*, vol. 17, pp. 25–45, 1997.

[2] P. Pajunen and J. Karhunen, "Least-squares methods for blind source separation based on nonlinear PCA," *Int. J. of Neural Systems*, vol. 8, pp. 601–612, 1998.

[3] X. L. Zhu and X. D. Zhang, "Adaptive RLS algorithm for blind source separation using a natural gradient," *IEEE Signal Processing Letters*, vol. 9, pp. 432–435, 2002.

[4] S. Haykin, *Adaptive Filter Theory (Third Edition)*, NJ: Prentice Hall, 1996.

[5] J. F. Cardoso and B. Laheld, "Equivariant adaptive source separation," *IEEE Trans. Signal Processing*, vol. 44, pp. 3017–3029, 1996.

[6] J. Karhunen, E. Oja, L. Wang, and et al., "A class of neural networks for independent component analysis," *IEEE Trans. Neural Networks*, vol. 8, pp. 486–504, 1997.

[7] J. Karhunen, P. Pajunen, and E. Oja, "The nonlinear PCA criterion in blind source separation: relations with others approaches," *Neurocomputing*, vol. 22, pp. 5–20, 1998.

[8] M.G. Jafari, H.W. Seah, and J.A. Chambers, "A combined kalman filter and natural gradient algorithm approach for blind separation of binary distributed sources in time-varying channels," *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, pp. 2769–2772, May 2001.

[9] J. Karhunen and J. Joutsensalo, "Representation and separation of signals using nonlinear PCA type learning," *Neural Networks*, vol. 7, pp. 113–127, 1994.

[10] B. Yang, "Projection approximation subspace tracking," *IEEE Trans. Signal Processing*, vol. 43, pp. 95–107, 1995.

[11] S. Amari, "Natural gradient works efficiently in learning," *Neural Computation*, vol. 10, pp. 251–276, 1998.

[12] M. Girolami, *Self-organising neural networks:independent component analysis and blind source separation*, Springer-Verlag London Berlin Heidelberg, 1999.

[13] H. Sawada, R. Mukai, S. Araki, and S. Makino, "A robust and precise method for solving the permutation problem of frequency-domain blind source separation," *IEEE Trans. on Speech and Audio Processing*, vol. 12, pp. 530–538, 2004.