

一种基于四阶累积量的改进快速 ICA 算法

蹇 柯, 崔志涛

(东莞理工学院城市学院 计算机与信息科学系, 广东 东莞 5 3106)

摘要: 快速 ICA(独立分量分析: Independent Component Analysis)算法是目前非常流行的一种好算法. 以四阶累积量作为优化判据, 在分析批处理的固定点快速分离算法基础上, 对牛顿迭代法进行了一系列改进, 提出了一种新的独立分量分析算法, 计算机的仿真结果验证了该改进算法的有效性.

关键词: 盲信号分离; 四阶累积量; 快速独立分量分析

中图分类号: TN911. 7

文献标识码: A

文章编号: 1007-716 (008) 04-0056-05

独立分量分析 (Independent Component Analysis ICA)是信号处理领域的一种新处理方法, 它与盲信源分离 (Blind Signal Separation, BSS)技术是紧密联系的. 在统计独立性的假设下, 独立分量分析是对由独立信源产生且经过未知线性混合的观测信号进行盲信源分离, 从而重现原独立信源. 它是根据盲分离技术能提取独立成分的特点而发展起来的, 其应用主要集中在盲信源分离和特征提取. 目前, 一些学者用 ICA 在人脸识别、遥感图处理、心电信号去噪、目标增强等方面取得的应用成果也充分展示了 ICA 广泛的应用前景.

图 1 为 ICA 的简单线性模型, 没有考虑含噪声

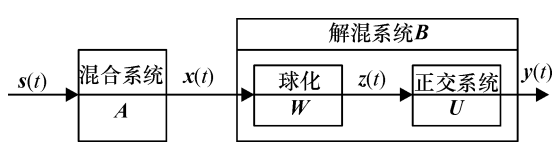


图 1 ICA 混合 - 解混线性模型简释

的情况. 用数学语言来描述 ICA 问题^[1]: 设有 N 个未知的源信号 $s_i, i = 1, \dots, N$, 构成一个列向量 $\mathbf{s} = [s_1, s_2, \dots, s_N]^T$. $\mathbf{A}$ 是一个 $M \times N$ 维源信号混合矩阵, $\mathbf{x} = [x_1, x_2, \dots, x_M]^T$ 是 M 维随机观测信号向量, 且满足下列方程

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t). \tag{1}$$

式 (1) 可写成矩阵形式, 即

$$\begin{bmatrix} x_1(t) \\ \vdots \\ x_M(t) \end{bmatrix} = \begin{bmatrix} a_{11} & \cdots & a_{1N} \\ \vdots & \ddots & \vdots \\ a_{M1} & \cdots & a_{MN} \end{bmatrix} \begin{bmatrix} s_1(t) \\ \vdots \\ s_N(t) \end{bmatrix}. \tag{2}$$

式 (2) 中, 每个混合信号 $x_i(t) (i = 1, \dots, M)$ 都可以

是一个随机信号, 其每个观测值 $x_i(t)$ 是在 t 时刻对随机信号 x_i 的一次抽样. 由式 (2) 可以看出, t 时刻各观测数据 $x_i(t)$ 是由各独立源信号 $s_j(t)$ 的值经过不同 a_{ij} 线性加权得到的.

ICA 的任务是寻找一个解混矩阵 $\mathbf{B}$, 使得

$$\mathbf{y}(t) = \mathbf{B}\mathbf{x}(t). \tag{3}$$

要求输出信号 y_i 相互独立, 则 $\mathbf{y} = [y_1, y_2, \dots, y_N]^T$ 就是 $\mathbf{s}$ 的估计值.

在信源 $\mathbf{s}(t)$ 中各分量互相独立的假设下, 由观测 $\mathbf{x}(t)$ 通过解混系统 $\mathbf{B}$ 将源信号分离出来, 使输出 $\mathbf{y}(t)$ 逼近 $\mathbf{x}(t)$. 一般解混系统 $\mathbf{B}$ 分两步进行处理: 第 1 步“球化”, 使输出 $\mathbf{z}(t)$ 的各分量 $z_i(t)$ 的方差为 1 而且互不相关 (但未必独立); 第 2 步“正交变换”, 一方面使输出 y_i 的方差保持为 1, 同时使各 y_i 尽可能相互独立. 此时不但 $p(\mathbf{y}) = \prod_{i=1}^M p(y_i)$, 而且协方差阵 $\mathbf{C}_y = E[\mathbf{y}\mathbf{y}^T] = \mathbf{I}_M$.

ICA 是在某一判据意义下进行的寻优计算, 它实际上包含两部分的内容: 采用什么判据作为一组信号是否接近相互独立的准则; 其次是用怎样的算法来达到这个目标.

虽然基本的快速 ICA 算法能较好地分离出源信号, 但它的收敛速度还不够快. 本文以四阶累积量作为优化判据, 在分析采用批处理的固定点快速分离算法的基础上, 利用修正牛顿迭代法的三阶收敛性, 减少求雅可比矩阵的次数, 提出了一种新的独立分量分析算法, 不仅也能较好地分离出源信号, 而且提高了收敛速度, 并通过计算机的仿真验证了算法的有效性.

1 四阶累积量判据

由中心极限定理知非高斯程度可以作为逐次提取独立分量 (Independent Component C) 时独立程度的判据, 而负熵应是最适当的衡量信号非高斯程度的度量.

本文中采用四阶累积量作为负熵的判据^[3]为 $J(y_i) = |k_4(y_i)|$. 因为

$$k_4(y_i) = E(y_i^4) - 3E^2(y_i^2),$$

且 $y_i = \mathbf{u}_i^T \mathbf{z}$ 所以 $E(y_i^2) = \mathbf{u}_i^T E(\mathbf{z}\mathbf{z}^T) \mathbf{u}_i = \mathbf{u}_i^T \mathbf{u}_i = \|\mathbf{u}_i\|^2$ (由于 $\mathbf{z}$ 是球化数据, 所以 $\mathbf{u}_i^T \mathbf{z} = 1$).

这样, $k_4(y_i)$ 可以改记为

$$k_4(\mathbf{u}_i^T \mathbf{z}) = E(y_i^4) - 3\|\mathbf{u}_i\|^4,$$

而 $J(y_i) = |k_4(\mathbf{u}_i^T \mathbf{z})| = |E[(\mathbf{u}_i^T \mathbf{z})^4] - 3\|\mathbf{u}_i\|^4|$. 应用梯度法时,

$$\Delta \mathbf{u}_i \propto \frac{\partial J(y_i)}{\partial \mathbf{u}_i} = \frac{\partial}{\partial \mathbf{u}_i} \left\{ |E[(\mathbf{u}_i^T \mathbf{z})^4] - 3\|\mathbf{u}_i\|^4| \right\} = 4 \operatorname{sgn}[k_4(\mathbf{u}_i^T \mathbf{z})] \left\{ E[\mathbf{z}(\mathbf{u}_i^T \mathbf{z})^3] - 3\mathbf{u}_i \|\mathbf{u}_i\|^2 \right\}.$$

稳态时 $\Delta \mathbf{u}_i = 0$ 因而有

$$E[\mathbf{z}(\mathbf{u}_i^T \mathbf{z})^3] - 3\mathbf{u}_i = 0 \quad [\|\mathbf{u}_i\| = 1],$$

由此得固定点的两步迭代算式

$$\begin{cases} \mathbf{u}_i(k+1) = E\{\mathbf{z}[\mathbf{u}_i^T(k)\mathbf{z}]^3\} \\ \mathbf{u}_i(k+1) \leftarrow \frac{\mathbf{u}_i(k+1)}{\|\mathbf{u}_i(k+1)\|} \end{cases} \quad (4)$$

由中心极限定理可知, 非高斯性可以作为随机信号相互依赖的度量, 所以当非高斯性达到最大时, 表明已完成对各独立分量的分离. y 的非高斯性越强, $J(y)$ 的值越大, 即最大化非高斯性就是最大化的四阶累积量.

快速 ICA 算法

目前, 流行的 ICA 算法大致可以分为两大类: 一类是基于批运算某种相关的判据函数, 这些算法一般需要进行复杂的矩阵和向量运算, 如文献 [4-7]; 另一类是基于随机梯度方法的自适应算法, 主要问题是运算的收敛速度慢. FastICA 算法的一般步骤如下:

1) 对观测信号 $\mathbf{x}$ 进行去均值预处理.

2) 通过线性变化改变观测信号 $\mathbf{x}$ 得到一个新的白噪声化变量 $\mathbf{x}'$, 使得 $\mathbf{x}'$ 的各个分量不相关. 白噪声化使 $\mathbf{x}'$ 新的混合矩阵 $\mathbf{A}'$ 正交化. 白噪声化的过

程为

$$\mathbf{x}' = \mathbf{C}\mathbf{x} = \mathbf{C}\mathbf{A}\mathbf{s} = \mathbf{A}'\mathbf{s}$$

3) 用 FastICA 算法分离出 $\mathbf{s}$ 的各个分量.

(1) 初始化 $\mathbf{u}(0)$, 令其模等于 1, 置 $k=1$;

(2) $\mathbf{u}(k) = E\{\mathbf{x}[\mathbf{u}^T(k-1)\mathbf{x}]^3\} - 3\mathbf{u}(k-1)$, 期望值可由大量 $\mathbf{x}$ 向量的采样点的平均值计算出来;

(3) 用 $\|\mathbf{u}(k)\|$ 去除 $\mathbf{u}(k)$;

(4) 如果 $|\mathbf{u}^T(k)\mathbf{u}(k-1)|$ 不是足够接近 1, 那么令 $k=k+1$, 返回第 (2) 步, 否则输出 $\mathbf{u}(k)$.

算法最后给出的向量 $\mathbf{u}(k)$ 等于正交混合矩阵 $\mathbf{A}$ 中的一列, 在信号分离中意味着 $\mathbf{u}(k)$ 分离了其中的一个非高斯信号 $\mathbf{u}(k)^T \mathbf{x}(t)$, $t=1, \dots$ 等于其中一个源信号.

3 改进的 FastICA 算法

第 2 章中的第 (2) 步即是 FastICA 的迭代过程, 针对第 (2) 步, 本文提取的改进算法在不增加求雅可比矩阵次数的情况下, 减少收敛时的迭代次数, 从而也获得减少计算量、提高算法速度的效果.

基本的牛顿迭代法:

$$x_{n+1} = x_n - [f(x_n) / f'(x_n)], \quad n = 0, 1, \dots \quad (5)$$

是求解非线性方程 $f(x) = 0$ 的重要方法. FastICA 算法就是根据式 (5) 得出的, 进而估算出 $\mathbf{U}$. 可以证明当 $f(a) = 0$ 但 $f'(a) \neq 0$ 时, 式 (5) 是二阶收敛. 对牛顿迭代法进行修改^[8]:

$$\begin{aligned} x_{n+1}^* &= x_n - [f(x_n) / f'(x_n)], \\ x_{n+1} &= x_n - [f(x_n) / (f'(x_n) + f'(x_{n+1}^*))], \\ n &= 0, 1, \dots \end{aligned} \quad (6)$$

修正后的牛顿迭代法具有三阶收敛, 这无疑加快了收敛速度. 下面给出修正牛顿迭代法三阶收敛性的证明.

定义 令 $x_0, x_1, x_2, \dots$ 是收敛于 a 的序列, 并令 $e_n = x_n - a$. 如果存在着数 p 及常数 $C \neq 0$ 使得 $\lim |e_{n+1}| / |e_n|^p = C$, 则称 p 为序列的收敛阶, 称 C 为渐进误差常数. 对于 $p=1, 3$ 分别称为线性、二阶、或者三阶收敛, 也称 $e_{n+1} = C e_n^p + O(e_n^{p+1})$ 为误差方程^[9].

设 $f(x): I \rightarrow \mathbf{R}$ 在开区间 I 内具有直到四阶的导数, 如果 $a \in I$ 是 $f(x) = 0$ 的单根, 且 x_0 充分靠近 a , 则由式 (6) 定义的修正牛顿迭代法具有三阶收敛, 且误差方程为

$$e_{n+1} = \left[C + \frac{1}{3}C_3 \right] e_n^3 + o(e_n^3), \quad (7)$$

其中 $e_n = x_n - a$, $C_k = (1/k!) f^{(k)}(a) / f'(a)$, $k=2, 3$

证明 将 $f(x)$ 在 a 处作 Taylor 展开, 则

$$\begin{aligned} f(x_n) &= f(a + e_n) = f(a) + f'(a)e_n + \\ &\quad \frac{1}{2!} f''(a)e_n^2 + \frac{1}{3!} f'''(a)e_n^3 + o(e_n^3) = \\ &\quad f'(a)[e_n + C_1 e_n^2 + C_2 e_n^3 + o(e_n^3)], \end{aligned}$$

$$f'(x_n) = f'(a)[1 + C_1 e_n + C_2 e_n^2 + o(e_n^2)].$$

对于式 (6), 则可得

$$\begin{aligned} x_{n+1}^* &= x_n - \frac{f(x_n)}{f'(x_n)} = a + C_1 e_n - (C_1 - C_2)e_n^2 + o(e_n^2), \\ f'(x_{n+1}^*) &= f'(a)[1 + C_1 e_n - 4(C_1^2 - C_1 C_2)e_n^2] + o(e_n^2), \\ x_{n+1} &= x_n - \frac{f(x)}{f'(x_n) + f'(x_{n+1}^*)} = a + \left(C_1 + \frac{1}{2}C_2\right)e_n^2 + o(e_n^2), \\ e_{n+1} &= \left(C_1 + \frac{1}{2}C_2\right)e_n^2 + o(e_n^2). \end{aligned}$$

即证明了由式 (6) 定义的修正牛顿迭代法具有三阶收敛, 且误差方程为

$$e_{n+1} = \left(C_1 + \frac{1}{2}C_2\right)e_n^2 + o(e_n^2).$$

根据修正的牛顿迭代法, 本文提出了基于四阶累积量的改进 FastICA 算法, 其具体表达形式为

$$\begin{aligned} \mathbf{u}_i^*(k) &= \mathbf{u}_i(k-1)E\{\mathbf{z}[\mathbf{u}_i^T(k-1)\mathbf{z}]^3\} - \\ &\quad 3E\{[\mathbf{u}_i^T(k-1)\mathbf{z}]\mathbf{u}_i(k-1)\}, \\ \mathbf{u}_i^*(k) &= \mathbf{u}_i^*(k)/\|\mathbf{u}_i^*(k)\| \\ \mathbf{u}_i(k) &= \mathbf{u}_i(k-1)E\{\mathbf{z}[\mathbf{u}_i^T(k-1)\mathbf{z}]^3\} - \\ &\quad 3E\{[\mathbf{u}_i^T(k-1)\mathbf{z}]\mathbf{u}_i(k-1)\} + E\{[\mathbf{u}_i^*(k-1)\mathbf{z}]\mathbf{u}_i(k-1)\}. \end{aligned} \quad (8)$$

本文提出的改进算法的具体步骤为

- 1) 对观测信号 $\mathbf{x}$ 进行中心化处理;
- 2) 对中心化后的 $\mathbf{x}$ 进行白化处理;
- 3) 初始化随机权矢量 $\mathbf{u}_0$, 设置收敛误差

$0 < \varepsilon \ll 1$;

4) 按照式 (8) 调整 $\mathbf{u}$ 计算出 $\mathbf{u}_k$ 后, 进行去相关和归一化;

5) 如果 $\|\mathbf{u}_k - \mathbf{u}_{k-1}\| < \varepsilon$ 则算法收敛, 估算出一个独立分量, 否则重复步骤 4) 和 5);

6) 由分离矩阵 $\mathbf{U}$ 得到源信号的所有估计 $\mathbf{y}_s$, 输出估计信号 $\mathbf{y}$.

4 仿真实验和算法分析

分别用 FastICA 算法和本文中的改进算法对波形信号和语音信号进行实验.

实验一 波形信号的盲分离: 选择正弦波、锯齿波、奇异曲线波和脉冲噪声波为 4 个波形源信号 (由 Matlab 提供, 见图 3), 随机产生一个 4×4 混合矩阵 $\mathbf{A}$ 对波形信号进行混合. 混合信号的波形如图 4

$$\mathbf{A} = \begin{bmatrix} 0.4864 & 0.0185 & 0.7919 & 0.4057 \\ 0.9501 & 0.8913 & 0.814 & 0.918 \\ 0.311 & 0.761 & 0.447 & 0.738 \\ 0.6068 & 0.4565 & 0.6154 & 0.1763 \end{bmatrix}$$

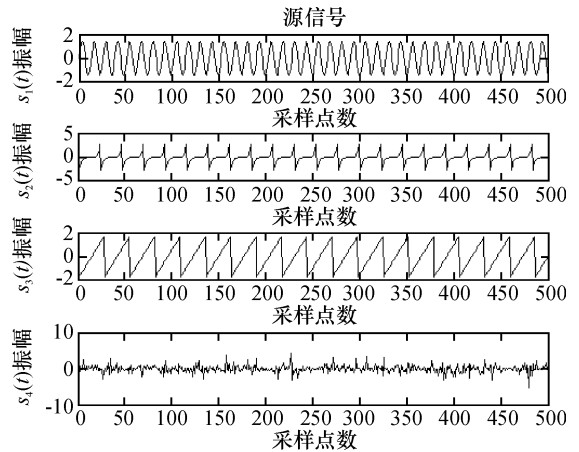


图 3 4 个波形源信号

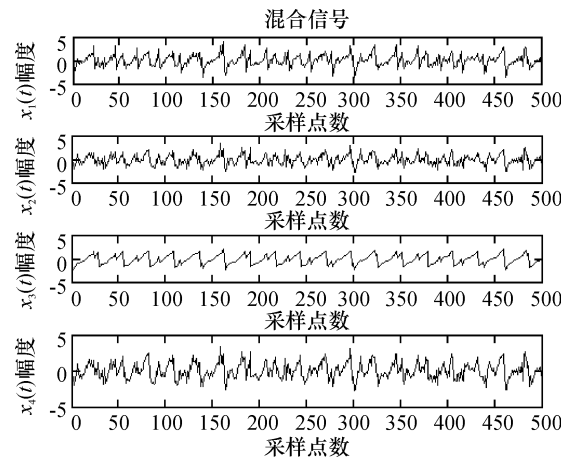


图 4 4 个波形混合信号

图 4 为基本 FastICA 算法分离的信号图, 图 5 为本文改进算法分离的信号图. 由分离的结果可见, 两种算法均能较好分离出源信号, 并未造成信号丢失.

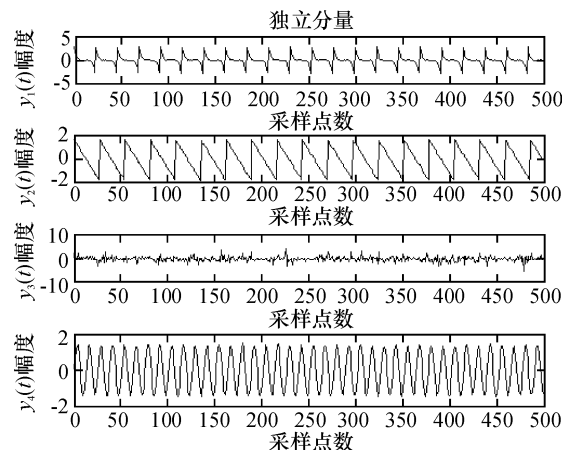


图 5 FastICA 算法分离的波形信号

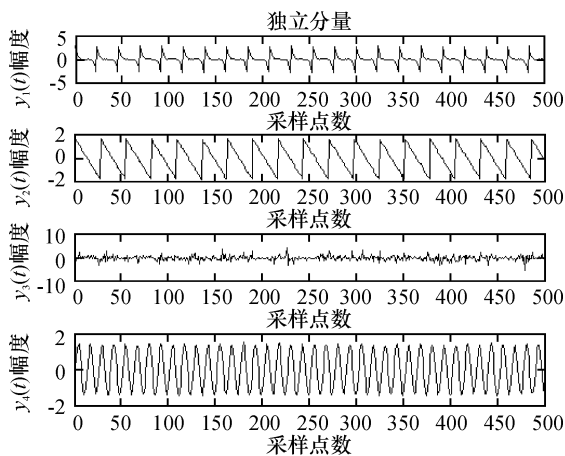


图 5 本文算法分离的波形信号

下面对 FastICA 算法和本文的改进算法的收敛速度进行比较, 分别列出分离每个分量 (Independent Component IC) 的迭代次数、每次的总迭代次数和平均迭代次数, 结果见表 1(为了更客观地比较两种算法, 每种算法均随机进行 10 次盲分离)。

表 1 波形信号收敛速度比较

序号	基本 FastICA 算法					改进 FastICA				
	IC ₁	IC ₂	IC ₃	IC ₄	总迭代次数	IC ₁	IC ₂	IC ₃	IC ₄	总迭代次数
1	9	7	6		4	9	7	5		3
	8	6	7		3	7	5	4		18
3	6	5	9			6	6			0
4	8	7	1		9	7	5	6		0
5	7	8	5			8	7	9		6
6	10	8	5		5	10	6	6		4
7	6	6	8			8	6	7		3
8	7	1	5		6	7	5	7		1
9	8	6	8		4	7	7	3		19
10	7	5	4		18	7	6	6		1
平均	7.6	7	6.9		3.5	7.6	6.7	5.9		1.5

从表 1 中可见, 本文改进算法 10 次盲分离的总迭代次数少于基本的 FastICA 算法, 经计算迭代速度提高了 8 51%, 从而说明了本文改进算法的有效性。

实验二 语音信号的盲分离: 两个语音信号来自 www.cis.hut.fi/projects/ica/, 语音源信号的波形图如图 6 所示。

图 7 为两个语音信号的混合图, 图 8 为 FastICA 算法分离的语音信号图, 图 9 则是本文算法分离的语音信号图。由分离结果可知, 两种算法均能较好地分离出源信号。下面对 FastICA 算法和本文的改进算法的收敛速度进行比较, 分别列出分离每个分量 (Independent Component IC) 的迭代次数、每次的总迭代

次数和平均迭代次数, 结果见表 (为了更客观地比较两种算法, 每种算法均随机进行 10 次盲分离)。

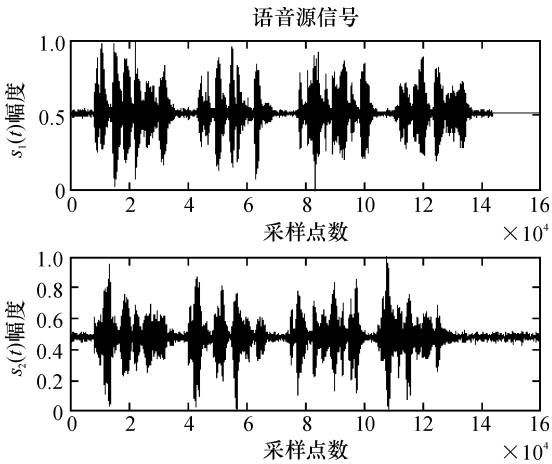


图 6 两个语音源信号

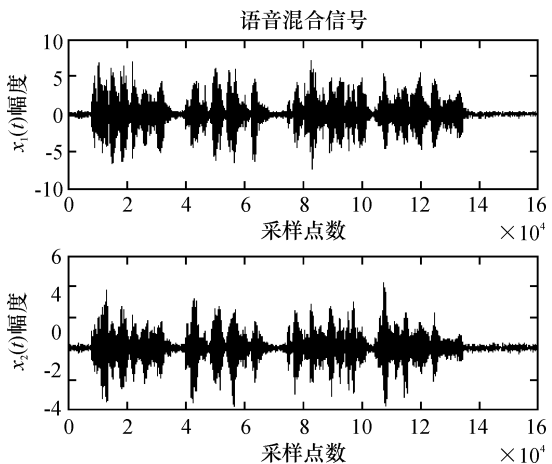


图 7 两个语音混合信号

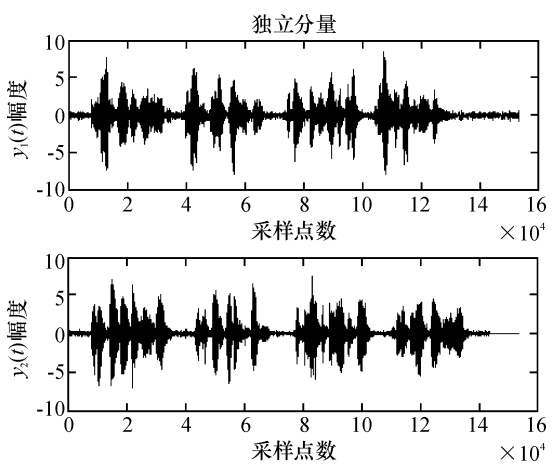


图 8 FastICA 算法分离的语音信号

从表 中可见, 本文的改进算法 10 次盲分离的总迭代次数少于基本的 FastICA 算法, 经计算迭代速度提高了 5 56%。

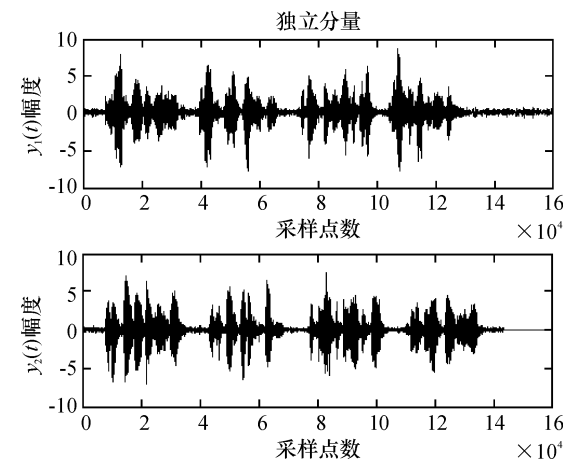


图 9 本文算法分离的语音信号

表 语音信号收敛速度比较

序号	基本 FastICA 算法		改进 FastICA	
	IC ₁	IC	IC ₁	IC
1	11	3	5	7
2	6	8	6	8
3	7	9	6	8
4	7	9	7	9
5	6	8	6	8
6	5	7	7	9
7	8	10	6	8
8	7	9	7	9
9	6	8	7	9
10	7	9	8	10
平均	7.0	9.0	6.5	8.5

5 结论

本文在对牛顿迭代法进行修正的基础上,提出了

一种改进 FastICA 算法,在与基本 FastICA 算法分离效果相当的情况下,在一定程度上减少了迭代次数,加快了收敛速度,结论的有效性得到了验证.但是针对图片信号或其他高维信号的混合盲分离,该改进算法的有效性还需要进一步验证,还需要具体作形式上的修改,这也是下一步讨论的问题.

参考文献:

[1] Comon P. Independent component analysis: A new concept [J]. Signal Processing, 1994, 36(3): 87-314

[2] Amari S, Cichocki A, Yang H H. A new learning algorithm for blind signal separation [J]. Advances in Neural Information Processing Systems, 1996, 8: 757-763

[3] Yang H H, Amari S, Shun-ichi. Adaptive on-line learning algorithm for blind separation: maximum entropy and minimum mutual information [J]. Neural Computation, 1997, 9(7): 1457-148.

[4] Hyvarinen A, Erkki O ja. Independent component analysis algorithm and application [J]. Neural Networks, 2000, 13(4-5): 411-430

[5] Hyvarinen A, Erkki O ja. A fast fixed-point algorithm for independent component analysis [J]. Neural Computation, 1997, 9(7): 1483-149.

[6] 刘海林. 基于广义特征值的病态混叠盲源分离算法 [J]. 电子学报, 2006, 34(11): 07 - 075.

[7] 刘海林, 谢胜利. 非线性盲源分离的多目标进化算法 [J]. 系统工程与电子技术, 2005, 7(9): 1576-157.

[8] 张荣, 薛国民. 修正的三次收敛的牛顿迭代法 [J]. 大学数学, 2005, 1(1): 80-8.

[9] Gemund Dahlquist. 数值方法 [M]. 包雪松, 译. 北京: 高等教育出版社, 1990

An Improved Fast-ICA Algorithm Based on Fourth-Order Cumulant

Jian Ke, Cui Zh+tao

(Department of Computer and Information Science, City College of Dongguan University of Technology, Dongguan 5 3106, China)

Abstract Fast-ICA method is a popular good algorithm nowadays. It proposes a new independent component analysis algorithm which uses fourth-order cumulant as the optimized criterion, and improves Newton iteration method by analyzing batch processing fixed-point fast-ICA method. The validity and performance of the proposed algorithm are verified by extensive computer simulations.

Key words blind signal separation, fourth-order cumulant, fast-ICA