

引入松弛因子的高阶收敛 FastICA 算法

季 策¹, 王艳茹¹, 沙明博², 杨正义³

(1. 东北大学 信息科学与工程学院, 辽宁 沈阳 110819; 2. 奥维通信股份有限公司, 辽宁 沈阳 110179;

3. 中国人民解放军 94816 部队, 福建 福州 350002)

摘 要: 高阶收敛的 FastICA 算法对初始值的选择较为敏感, 如果初始值选择不当不仅会影响算法的收敛效果, 甚至可能导致不收敛的结果. 针对这一问题, 将松弛因子引入高阶收敛的牛顿迭代法中, 通过适当的修正, 获得了既能保证一定收敛速度, 又能有效克服初值敏感性的改进三阶、五阶 FastICA 算法. 仿真工具采用 Matlab 软件, 应用 3 种算法对语音信号进行分离; 结果表明, 对比基本 FastICA 算法, 改进后的算法有效地分离了混合信号, 并且降低了算法对初始权值的依赖性.

关 键 词: 独立分量分析; FastICA; 松弛因子; 初值敏感性

中图分类号: TN 911.6

文献标志码: A

文章编号: 1005-3026(2014)02-0204-04

Relaxation Factor-based FastICA with Higher Order Convergence

Ji Ce¹, WANG Yan-ru¹, SHA Ming-bo², YANG Zheng-yi³

(1. School of Information Science & Engineering, Northeastern University, Shenyang 110819, China; 2. Allwin Telecommunication Co., Ltd., Shenyang 110179, China; 3. 94816 PLA Troops, Fuzhou 350002, China.

Corresponding author: Ji Ce, associate professor, E-mail: jice@ise.neu.edu.cn)

Abstract: High order FastICA (fast independent component analysis) algorithm has the characteristics of simple form and quick convergence. However, the algorithm is sensitive to its initial value which affects the convergence effect and even results in non-convergence if it is not chosen appropriately. In order to solve this problem, a relaxation factor is introduced into high order Newton iterative method. Through the appropriate correction, the improved high order FastICA algorithm can be obtained, which can not only guarantee the convergence speed, but also effectively overcome the initial value sensitivity problem. Applying the algorithm to the separation experiment of speech signals, the result shows that the proposed algorithm effectively separates the mixed signal, and reduces the dependence on the initial value.

Key words: independent component analysis (ICA); FastICA; relaxation factor; initial value sensitivity

独立分量分析是解决盲源分离问题的一种有效方法^[1], 其中的 FastICA 算法, 即所谓的固定点算法^[2], 又是独立分量分析中比较重要的优化算法, 被广泛应用于语音处理^[3]、特征提取、图像处理^[4]、人脸识别及生物医学^[5]等许多领域, 并且取得了令人满意的成果. FastICA 算法采用牛顿迭代法原理^[6], 具有收敛速度快的优点, 尤其是改进的高阶 FastICA 算法, 具有三阶^[7]乃至五阶^[8]的收敛速度. 然而, 在研究高阶 FastICA 算法

的过程中发现, 由于算法的初始权值往往是随机选取的, 所以, 选取不同的初始值常常会导致每次收敛时的迭代次数不同, 甚至可能会出现收敛、不收敛两种极端的情况, 从而影响分离效果. 这一问题, 在近年来一直没有得到很好的解决.

为了克服高阶 FastICA 算法的初值敏感问题^[9], 放宽算法对初始权值的依赖性, 本文在文献^[7-8]给出的高阶收敛的 FastICA 算法的基础上, 通过引入松弛因子, 对随机产生的初始权值进

行处理,进而实现算法的大范围收敛.语音信号的仿真实验证明:在求最优解的过程中,改进后的算法不再依赖于初始值的选择,很好地克服了初值敏感性问题,进而提高了高阶收敛的 FastICA 算法的分离性能.

1 基于松弛因子的高阶收敛 FastICA 改进算法

FastICA 算法是 A. Hyvarinen 提出的基于不动点迭代的盲信号分离算法,也称固定点算法.

传统的 FastICA 算法具有二阶收敛速度,文献[7-8]在传统 FastICA 算法的基础上加以改进,达到了三阶或五阶的收敛速度.

1.1 松弛因子对 FastICA 算法的影响

为降低算法对初值的敏感性,本文在文献[7-8]三阶和五阶收敛的 FastICA 算法的基础上,将松弛因子 α_k 引入其迭代公式中.

松弛因子 α_k 的引入应保证目标函数 $F(\mathbf{w}_k)$ 按给定范数具有下降的性质,即,使其在某种范数下满足:

$$\begin{aligned} & \|E\{xg(\mathbf{w}_{k+1}^T \mathbf{x})\} - \beta \mathbf{w}_{k+1}^T\| < \\ & \|E\{xg(\mathbf{w}_k^T \mathbf{x})\} - \beta \mathbf{w}_k^T\|, \\ & k=0,1,\dots, \end{aligned} \quad (1)$$

亦即

$$\min_{\alpha_k} \|F(\mathbf{w}_k - \alpha_k F(\mathbf{w}_k)/JF(\mathbf{w}_k))\|. \quad (2)$$

式中: $F(\mathbf{w}_k) = E\{xg(\mathbf{w}_k^T \mathbf{x})\} - \beta \mathbf{w}_k$, $\beta = E\{\mathbf{w}_i^T \mathbf{x}g(\mathbf{w}_i^T \mathbf{x})\}$; J 为雅可比矩阵; $g(\cdot)$ 为非线性函数; \mathbf{w} 为分离矩阵; \mathbf{x} 为观测矩阵; $E\{\cdot\}$ 为期望值.

这种下降性质可以促使 $F(\mathbf{w}_k)$ 从某个 \mathbf{w}_k 开始进入牛顿迭代法的收敛区域,确保了算法可以在任何情况下达到收敛的效果.当 $\alpha_k = 1$ 时,该算法则退化为普通的高阶收敛的 FastICA 算法.可以看出,引入松弛因子的牛顿迭代法放宽了初始权值的选择条件.尽管引入松弛因子在一定程度上影响了收敛速度,但由于高阶收敛的 FastICA 算法具有三阶和五阶的收敛速度,因此,加入松弛因子后的 FastICA 算法,可以在不影响收敛速度的前提下,克服初值敏感性问题.

算法动态地自动选取松弛因子 α_k . 首先,令 $\alpha_k = 1$, 然后求 $\Delta \mathbf{w}_k = F(\mathbf{w}_k)/JF(\mathbf{w}_k)$, 检查式(3)是否成立.

$$\|F(\mathbf{w}_k - \alpha_k \Delta \mathbf{w}_k)\|^2 < \|F(\mathbf{w}_k)\|^2. \quad (3)$$

如果式(3)不满足,则用 $0.5\alpha_k$ 代替 α_k , 循环检查,直至满足式(3)条件为止;令 $\mathbf{w}_{k+1} = \mathbf{w}_k -$

$\alpha_k \Delta \mathbf{w}_k$, 完成一次基于松弛因子的改进 FastICA 算法的迭代.

1.2 改进算法的实现

1.2.1 基于松弛因子的三阶收敛 FastICA 算法

$$\left. \begin{aligned} \mathbf{w}^\# &= \alpha_k E\{xg(\mathbf{w}^T \mathbf{x})\} - E\{g'(\mathbf{w}^T \mathbf{x})\} \mathbf{w} - \beta(1 - \alpha_k) \mathbf{w}, \\ \mathbf{w}^+ &= 2E\{xg(\mathbf{w}^T \mathbf{x})\} - E\{g'(\mathbf{w}^T \mathbf{x})\} + E\{g'(\mathbf{w}^\# \mathbf{x})\} \mathbf{w}, \\ \mathbf{w}^* &= \mathbf{w}^+ / \|\mathbf{w}^+\|. \end{aligned} \right\} \quad (4)$$

式中, $\beta = E\{\mathbf{w}_i^T \mathbf{x}g(\mathbf{w}_i^T \mathbf{x})\}$.

算法的实现步骤如下.

①对观测信号 \mathbf{x} 进行中心化及白化处理;

②对随机权矢量 \mathbf{w}_0 进行初始化,并设收敛误差 $\varepsilon = 1 \times 10^{-6}$;

③设 $\alpha_k = 1$, 求出 $\Delta \mathbf{w}_k$, 检查是否满足 $\|F(\mathbf{w}_k - \alpha_k \Delta \mathbf{w}_k)\|^2 < \|F(\mathbf{w}_k)\|^2$, 若不满足,则用 $0.5\alpha_k$ 代替 α_k , 循环检查,直至满足上式条件为止;

④按式(4)对 \mathbf{w}_{k+1} 进行调整,从而计算出 \mathbf{w}_{k+1} , 再利用 $\mathbf{w}_{k+1} = \frac{\mathbf{w}_{k+1}}{\|\mathbf{w}_{k+1}\|}$ 进行去相关及归一化;

⑤当 $\|\mathbf{w}_{k+1} - \mathbf{w}_k\| < \varepsilon$, 则认为算法是收敛的,估算出一个独立分量,如果不收敛,则循环执行步骤④直至算法收敛;

⑥通过分离矩阵 \mathbf{W} , 得到源信号的所有估计值 y_i , 最后输出估计信号 \mathbf{Y} .

1.2.2 基于松弛因子的五阶收敛 FastICA 算法

$$\left. \begin{aligned} \mathbf{w}^\# &= \alpha_k E\{xg(\mathbf{w}^T \mathbf{x})\} - E\{g'(\mathbf{w}^T \mathbf{x})\} \mathbf{w} - \beta(1 - \alpha_k) \mathbf{w}, \\ \mathbf{w}^+ &= 2E\{xg(\mathbf{w}^T \mathbf{x})\} - E\{g'(\mathbf{w}^T \mathbf{x})\} + E\{g'(\mathbf{w}^\# \mathbf{x})\} \mathbf{w}, \\ \mathbf{w}^\oplus &= E\{xg(\mathbf{w}^+ \mathbf{x})\} - E\{g'[\frac{1}{2}(\mathbf{w}^\# + \mathbf{w}^+) \mathbf{x}]\} \mathbf{w}^+, \\ \mathbf{w}^* &= \mathbf{w}^\oplus / \|\mathbf{w}^\oplus\|. \end{aligned} \right\} \quad (5)$$

式中, $\beta = E\{\mathbf{w}_i^T \mathbf{x}g(\mathbf{w}_i^T \mathbf{x})\}$.

算法实现步骤如同三阶算法(步骤④中按式(5)对 \mathbf{w}_{k+1} 进行调整).

2 仿真实验及性能分析

2.1 仿真实验

仿真实验是对语音信号进行分离,应用 Matlab 软件,选取了采样频率为 10 kHz、采样点数为 5×10^4 的两路语音信号.先对选取的两路语音信号(图 1a)按随机矩阵(6)进行混合,进而得到一组混合信号(图 1b).

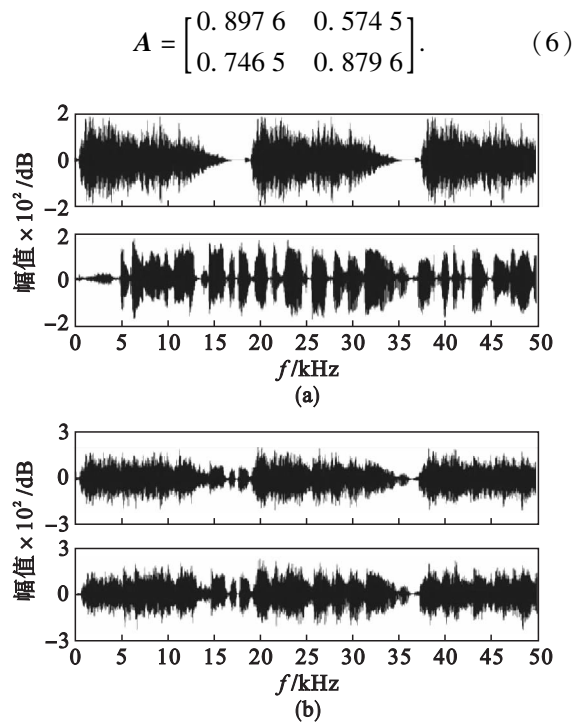


图 1 原始语音信号及其混合后语音信号
Fig. 1 The original signals and their mixed signals
(a)—两路原始语音信号；(b)—混合后语音信号.

分别用传统 FastICA 算法和改进的高阶收敛 FastICA 算法对混合后的两组语音信号进行分离处理,分离后的结果如图 2 所示. 其中,LF - FastICA - T 和 LF - FastICA - F 分别为基于松弛因子的三阶和五阶收敛的 FastICA 算法. 对比仿真图可以看出,改进后的算法成功地对语音信号进行了分离.

为了比较算法的收敛速度,分别对 3 种算法随机进行了 8 次分离实验,而在 8 次实验中均随机给定初始值,然后记录实验中各种算法的迭代次数,并计算每次实验的平均迭代次数.

观察表 1 可以看出,传统 FastICA 算法迭代次数在 9 ~ 13 之间波动;LF - FastICA - T 算法迭代次数在 8 ~ 9 之间波动;LF - FastICA - F 算法迭代次数则在 7 ~ 9 之间波动. 由此可见,改进的高阶收敛算法波动较小,不仅有效地降低了对初始值的依赖性,而且达到了收敛速度均衡的目的. 虽然在松弛因子的选取与计算过程中,可能会多消耗一些计算时间,然而由于改进算法是建立在三阶和五阶收敛的 FastICA 算法的基础上,因此和传统的 FastICA 算法相比,改进算法有更快的收敛速度.

从表 2 中可以看出,LF - FastICA - F 算法的运行时间比 FastICA 算法的短,即前者提高了收敛效率.

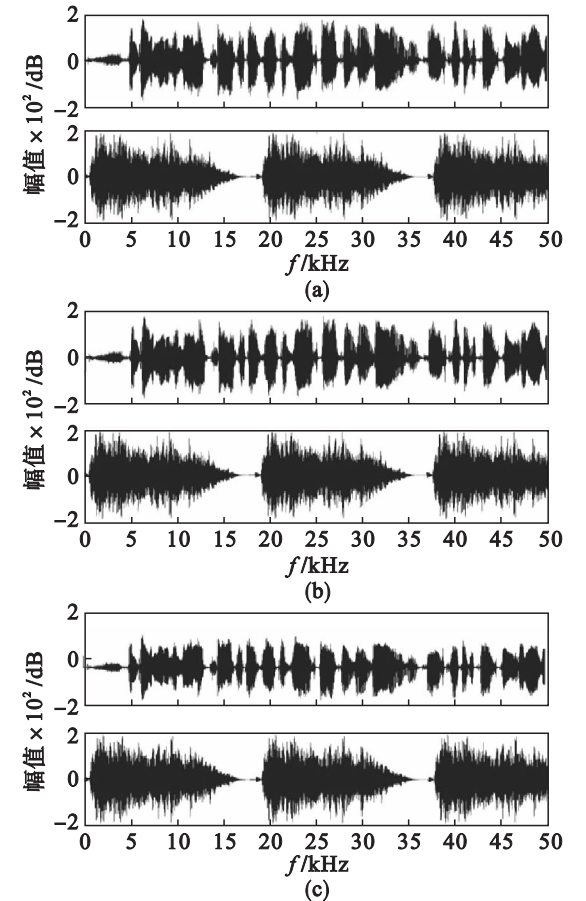


图 2 不同算法对语音信号的分离
Fig. 2 The signals separated with different algorithms
(a)—基本 FastICA 算法；(b)—LF - FastICA - T 算法；
(c)—LF - FastICA - F 算法.

表 1 迭代次数的比较									
Table 1 Comparison of the number of iterations									
算法	运算序号								平均 次数
	1	2	3	4	5	6	7	8	
FastICA	11	10	11	11	11	9	12	13	11
LF - FastICA - T	8	8	8	9	9	8	9	9	8.5
LF - FastICA - F	8	7	7	8	8	8	8	9	7.8

表 2 迭代时间的比较			
Table 2 Comparison of iteration time			
运算 序号	算法		
	FastICA	LF - FastICA - TT	LF - FastICA - F
1	0.343 23	0.313 45	0.286 84
2	0.354 65	0.313 34	0.283 33
3	0.381 11	0.311 58	0.284 58
4	0.329 85	0.314 58	0.285 68
5	0.384 42	0.312 24	0.291 26
6	0.343 55	0.316 69	0.283 39
7	0.371 25	0.325 58	0.296 75
8	0.380 11	0.326 59	0.294 56
平均 时间	0.361 02	0.316 75	0.288 29

2.2 性能分析

本文利用 PI 值^[10]来衡量算法的分离性能, PI 值计算如下:

$$PI = \frac{1}{2M} \sum_{i=1}^M \left\{ \left(\sum_{j=1}^M \frac{|c_{ij}|^2}{\max_k |c_{ik}|^2} - 1 \right) + \left(\sum_{j=1}^M \frac{|c_{ji}|^2}{\max_k |c_{ki}|^2} - 1 \right) \right\} \quad (6)$$

式中: M 代表变量个数; $\mathbf{C} = \mathbf{WA} = (c_{ij})$. PI 值越小,证明分离性能越好;PI = 0 时,代表信号已完全分离. 通过对语音信号的分离实验,计算得到各算法的 PI 值如表 3 所示.

表 3 PI 值的比较
Table 3 Comparison of PI values

算法	PI 值
FastICA	0.045 5
LF – FastICA – T	0.039 7
LF – FastICA – F	0.038 4

观察表 3,改进后的 LF – FastICA – T 算法和 LF – FastICA – F 算法的 PI 值均比传统 FastICA 算法小,由此说明改进的算法较好地保留了语音信号的信息.

3 结 语

传统牛顿迭代法存在对初值敏感的问题,为了克服这一缺点,本文基于三阶收敛和五阶收敛的 FastICA 算法,通过在算法中引入松弛因子,给出了改进的高阶 FastICA 算法. 语音信号的分离实验表明,改进算法有效地克服了初值敏感性问题,避免了收敛速度的不均衡,提高了算法的效率.

参考文献:

[1] Castano S, Ferrara A, Montanelli S, *et al.* A general framework for ontology-based knowledge sharing and a evolution in P2P systems [C]// The 14th International

Workshop on Database and Expert Systems Applications. Prague:IEEE Press,2003:597 – 603.

[2] Hyvarinen A, Karhunen J, Oja E. Independent component analysis[M]. Hoboken:John Wiley & Sons,2001.

[3] Asno F, Ikeda S, Ogawa M, *et al.* Combined approach of array processing and independent component analysis for blind separation of acoustic signals[J]. *IEEE Transactions on Speech and Audio Processing*,2003,11(3):204 – 215.

[4] 崔会丽,魏航,张军. 基于 FastICA 算法的高光谱图像混合像元分解[J]. 佳木斯大学学报:自然科学版,2012,30(5):751 – 753.

(Cui Hui-li, Wei Hang, Zhang Jun. Fast independent component analysis algorithm for mixed pixel decomposition of hyperspectral imagery[J]. *Journal of Jiamusi University: Natural Science Edition*,2012,30(5):751 – 753.)

[5] Cvejic N, Bull D, Canagarajah N. Region-based multimodal image fusion using ICA bases [J]. *IEEE Sensors Journal*, 2007,7(5):743 – 751.

[6] von Hoff T P, Lindgren A G, Kaelin A N. Step-size control in blind source separation[C]// International Workshop on Independent Component Analysis and Blind Signal Separation. Helsinki,2000:509 – 513.

[7] Ji C, Yu Y, Yu P. A new FastICA algorithm of Newton’ iteration [C]//The 2010 International Conference on Education Technology and Computer. Shanghai,2010:481 – 484.

[8] 季策,于洋,于鹏. 改进的独立分量分析算法[J]. 东北大学学报:自然科学版,2010,31(8):1086 – 1088.

(Ji Ce, Yu Yang, Yu Peng. Improved algorithm for independent component analysis[J]. *Journal of Northeastern University: Natural Science*,2010,31(8):1086 – 1088.)

[9] 季策,胡祥楠,朱丽春,等. 改进的高阶收敛 FastICA 算法 [J]. 东北大学学报:自然科学版,2011,32(10):1390 – 1393.

(Ji Ce, Hu Xiang-nan, Zhu Li-chun, *et al.* Improved higher order convergent FastICA algorithm [J]. *Journal of Northeastern University: Natural Science*, 2011, 32 (10): 1390 – 1393.)

[10] Paraschiv-Ionescu A, Jutten C, *et al.* Wavelet denoising for highly noisy source separation [C]//IEEE International Symposium on Circuits and Systems. Phoenix-Scottsdale, 2002:1201 – 1203.