ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A NOVEL APPROACH TO INDEPENDENT COMPONENT ANALYSIS

JAYASANTHI RANJITH [1], MUNIRAJ NJR [2]

[1] Research Scholar, Department of Information and communication Engineering, Anna university
Tamil nadu,India

[2] Principal, Tejaa Shakthi Institute of Technology for Women, Anna university
Tamil nadu, India

E-mail: [1] mjayasanthi@yahoo.co.in, [2] njrmuniraj@yahoo.com

ABSTRACT

Independent component analysis (ICA) is a computational method, based on neural learning algorithm, to separate source signals from the observed mixtures by assuming that the sources are non-Gaussian in nature. Convergence speed, Area and Power are important parameters to be improved in VLSI implementation of ICA techniques, since they involve large number of iterative calculations, area and power. This paper presents a novel fast confluence adaptive independent component analysis (FCAICA) technique for separation of signals from their two observed mixtures. The reduction in area and power is achieved by hardware optimization by replacing random generator unit by means of comparator. High convergence speed is achieved by a novel optimization scheme that adaptively changes the weight vector based on the kurtosis value. To increase the number precision and dynamic range of the signals, floating-point (FP) arithmetic units are used. Simulation, synthesis and backend analysis are carried out with ALTERA Quartus II Tool 11.1. The FCA ICA algorithm performs well for convergence time, maximum operating frequency, area and power. Also it is more effective compared with most popular FastICA and Evolutionary ICA algorithms.

Keywords: *Adaptive Independent component analysis,Contrast function optimization, Floating point,Very Large Scale Integration, Convergence speed.*

1. INTRODUCTION

A central problem in statistics and signal processing is to find suitable representation of data using a suitable transformation. A data driven method that does not require pre-specified response model to solve this problem is Independent component analysis (ICA).ICA recovers source signals from their mixtures by finding a linear transformation that maximizes the mutual independence or non-gaussianity of the mixtures regardless of the probability distribution. Kernal ICA which is a nonlinear function, developed in [1], maps signals to a high dimensional space to extract sources. Indeterminacy present in classical ICA technique is eliminated using some prior knowledge of source [2].

FastICA algorithm is the most popular algorithm among different reported ICA algorithms. This is because it has been shown to have advantage in terms of convergence speed .It measures the non-Gaussianity using kurtosis to find the independent sources from their mixtures [4]. Most ICA algorithms that are based on the maximum likelihood (ML) or the maximization of negentropy (MN) principle are equivalent when the demixing matrix is constrained to be unitary [5]. The most popular FastICA algorithm [6] uses the principle of maximization of negentropy and has the unitary constraint on the separating matrix. The simplest algorithm for maximizing the likelihood uses stochastic gradient methods. Another approach to ICA that is related to PCA is the non-linear method. Learning rule uses higher order information when nonlinearities are introduced. Algorithms for exactly maximizing the nonlinear PCA criteria are introduced in [7]. Using the principle of stochastic gradient descent, simple algorithms are derived from the one-unit contrast functions. Hebbian like learning rule is obtained by taking instantaneous gradient of contrast function with respect to w [8]. Joint approximate diagonalization of eigenmatrices (JADE) is based on the principle of computing several cumulant Tensors. With low dimensional data, JADE is a competitive alternative to most popular FastICA algorithms. Other approaches include maximization of squared cumulants[9] and fourth-order cumulant based methods [10].Fourth-

Journal of Theoretical and Applied Information Technology
20th May 2014. Vol. 63 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

order blind identification (FOBI) method deals with the Eigen value decomposition (EVD) of the weighted correlation matrix [11]. A frequency-domain method of blind source separation (FD-BSS) is able to separate acoustic sources under highly reverberant challenging conditions [12]. In frequency-domain BSS, the separation is generally performed by applying ICA at each frequency envelope. ICA is also done by entropy bound minimization (ICA-EBM) [13].

Population search based Evolutionary optimization techniques like genetic algorithms, particle swarm optimization are used in ICA [14].The only disadvantage of evolutionary computation based ICA technique is that it has heavy computational complexity. But with the advantage of highly parallel processors and new technologies like Very Large Scale Integration (VLSI), these methods provide competitive solutions to the problems. Fixed-point VLSI architecture was proposed for 2-Dimensional Kurtosis optimization based FastICA with reduced and optimized arithmetic units [16][17]. The Floating Point (FP) Number System is one of the good alternatives for applications that require an increased dynamic range for their arithmetic operations.

Parallel ICA (pICA) partitions ICA module into three temporally independent functional modules and synthesizes each of them individually. It provides optimal parallelism environment, a potential faster and real-time solution [18]. FPGA implementation of ICA in digital chip is reported with modular design concept [19] and with systolic architecture [20]. FPGA implementation of 32-channel convolutive ICA chip is reported for real world signal separation [22]. Pipelined FastICA, which can process the real time sequential mixed signal, is also developed for FPGA implementation [23]. Various analog VLSI implementations of ICA also exist in the literature. Since digital adaptation offers the flexibility of reconfigurable ICA, digital implementations are common in signal processing. Though there are softwares to translate the high-level languages such as C code, MATLAB, and even Simulink into HDL code, hand coding gives the optimized solution to the problems.

In this paper, Fast Confluence Adaptive ICA is proposed in floating point arithmetic with improved convergence speed. The most commonly used Fast ICA algorithm and Shuffled frog leap optimization based ICA algorithm are also developed for comparison purpose.

1.1 Contribution

The originality of the proposed FCAICA is summarized as Follows:

- Convergent speed is improved by adaptively changing the weight vectors according to the properties of fitness function
- The early determination of converging weight vector and demixing matrix reduces many operations and hence the power consumption.
- Floating point arithmetic improves the precision and dynamic range of the signals. It also enables VLSI implementation of real time signal processing.
- The proposed algorithm has been validated for subgaussian and supergaussian mixtures.

2. FLOATING POINT ARITHMETIC AND ICA BACKGROUND

2.1 Floating Point Arithmetic

Signal processing is done in two categories of numbers. One is based on fixed point arithmetic and another one is based on floating point arithmetic. Since most of the Signal processing techniques necessitate large dynamic signal range to achieve accuracy, fixed point representation provides unsatisfactory results. IEEE single precision format, that uses 32 bits, has been used for the proposed ICA algorithm. The 32 bit Floating Point number (F) is represented by (1)

$$F= (-1)^S \text{ X } 2\, E^{-127} \text{ X } (1.M) \quad (1)$$

The sign field $(-1)^S$ is used to specify the sign of the real number. Exponent field $(2\, E^{-127})$ is represented by using a bias of 127. It is a 8 bit quantity. The third field (1.M) is normalized binary significand with a hidden integer bit 1. Since leading one in the mantissa is implicit, it does not appear in the representation. Addition, multiplication, subtraction, division and square root operations are carried out following the appropriate algorithms of single precision IEEE 754 standard.

2.2 Background Of ICA

Blind source separation is a problem of finding a linear representation of hidden data from mixture in which the components are statistically independent. In practical situations, it is not possible to find a representation where the components are really independent, but it is possible to find components that are at least as independent as possible. The relationship between source signals S and observed mixtures X is given in matrix notation as in (2).

$$X=A.S \quad (2)$$

Journal of Theoretical and Applied Information Technology
20th May 2014. Vol. 63 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A is a full rank matrix that is called mixing matrix. Under some assumptions, ICA solves the BSS problem by finding inverse linear transformation such that, it maximizes the statistical independence between the observed mixtures. For doing this, ICA finds demixing matrix B which is inverse of mixing matrix A. Then the estimate of the source signal (S_est) is found from (3)

S_est=B.X (3)

i.e when a mixed signal(X) is multiplied with inverse of mixing matrix , estimate of the original signal (S_est) can be found. The demixing matrix B is found after performing preprocessing steps centering and whitening.

3. EVOLUTIONARY ICA (SFLOICA)

Shuffled Frog Leap optimization based ICA (SFLOICA) is an evolutionary ICA algorithm that performs contrast function optimization based on most popular Shuffled frog Leap optimization technique. Due to its increased search space, optimality performance is improved. Mutation operator and crossover operator are introduced in this algorithm to avoid the solution from getting trapped in local minima and to move the solution towards global minima.

In this algorithm, initial weight vectors for estimating the demixing matrix are assumed as frogs and updated by step 3 of the algorithm. Then fitness value is calculated and sorting is done according to fitness value. Based on the fitness values, total population is partitioned into q groups (memeplexes) of p frogs that search independently. In this process, the first frog goes to the first memeplex, the second frog goes to the second memeplex, frog p goes to the pth memeplex, and frog p +1 goes back to the first memeplex and so on. In each memeplex, the frogs with the best and the worst fitnesses are identified as Xb and Xw respectively. Also, the frog with the most qualified fitness level among all the memeplexes is identified as Xg. Then improvement is done to improve only the frog with the worst fitness according to step (3). If this process produces a better solution, it replaces the worst frog. Otherwise, a new population is randomly generated to replace that population. This process continues for a specific number of iterations (Imax1).Then all memeplexes are combined and sorted. Then mutation operation is included using step 11 to avoid local minima. If the current iteration number reaches (Imax2), the search procedure is stopped; otherwise it goes to Step 3. The last Xg is the solution of the problem.

3.1 Floating Point Iteration

Estimation of w(k+1) is done with pre-processed data iteratively using following steps until a convergence is achieved.

1) Choose initial population of 'n' frogs (weights) at random.
2) Find norm of pair of frogs and divide by corresponding norms.
3) Update the frogs by the formula
   $$w(k+1) \leftarrow E\{Z(w(k)Z^T)^3\} - 3w(k)$$
4) Calculate the fitness value from f=w(k+1)-w(k)
5) Sort the initial population based on the fitness values.
6) Partition the sorted population into p memeplexes of q frogs.
7) Select the best frog Xb, worst frog Xw in each memeplex and globally best frog Xg.
8) Update the position of Xw using
   Xw (new) =Xw(old) +C
   where C= rand( ).(Xb-Xw).
9) If it produces better solution, older frog is replaced by updated frog and this process continues for a specific number of iterations (Imax1). Otherwise a new frog is randomly generated to replace Xw and algorithm goes to step 2.
10) All memeplexes are combined and sorted again.
11) Apply mutation
12) If the current iteration number reaches Imax2, the search procedure is stopped or it goes to Step 5.
13) The last Xg is the solution of the problem.

With the above steps, Deflationary othogonalization is made to find remaining independent components.

The disadvantage of this method is that, the operations like crossover, mutation and sorting increases the complexity of the algorithm. So a novel algorithm with reduced complexity without affecting the optimality of the solution is developed and discussed with results in the next section.

4. NOVEL FAST CONFLUENCE ADAPTIVE ICA

4.1 FCAICA Algorithm

Having done the preprocessing to whiten the mixed signal, this algorithm finds the independent components for extraction of desired signal from the mixtures. The proposed Fast Confluence

Journal of Theoretical and Applied Information Technology
20th May 2014. Vol. 63 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Adaptive ICA algorithm for one unit estimates one row of the demixing matrix. Updation of weights continues in iterative manner with the following steps until a convergence is achieved.

1) Form a weight matrix W by assuming N sub matrices or column vectors as

W={ w1,w2,w3,w4,…………wN}
where wi=(wi1,wi2……………..,win)T
wij= jth weight of ith column vector
n= Number of sources

N defines the size of the search space to obtain globally best solution.

2) Find norm of each vector and divide by corresponding norms.

3) Find the updated weight vector W(k+1) for all the weights in W using (4)

$$w(k+1) \leftarrow E\{Z(w(k)Z^T)^3\} - 3w(k) \quad (4)$$

4) Calculate the fitness value Φi for all weights

Φi= wi(k+1)-wi(k) (5)

5) Divide the N weight vectors into M sorted groups (N=2*M) with 2 (n=2) weights in each group. The division is done in such a way that 1st weight goes to 1st group, 2nd weight goes to 2nd group and continuous up to M weight. Then (M+1)th weight goes to 1st group and so on.

6) In each group , If Φi (k)< Φi (k+1) then
refi(k)= Φi (k); (6)
else
refi(k)= Φi (k+1) (7)
end

7) If refi (k) is positive
wi(k)= wi(k)-C (8)
else
wi(k)=wj(k)+C; (9)
end

Where C is a random nonnegative floating point number between '0' and '1'.

8) Repeat from step 2 to find wi(k+1) until convergence is achieved.

The convergence is checked by examining the direction of old and new weight vectors. When both vectors point at the same direction convergence is proved. It is known that the convergence speed of fast ICA is cubic or atleast quadratic.

Based on the fitness value of each assumed weight vector, the direction of the desired search space is found. This direction decides whether searching is to be done either towards left or towards right and thus the search space(N) is reduced.As the search space has been reduced, the convergence time has also been reduced without affecting the quality of the optimal solution. The floating point operations enhance the quality of optimal solution.

Once the convergence is achieved, the two vectors with good fitness value are used as row vectors of demixing matrix B. This B is then used to find the estimate of source signal. While finding more than one independent component, Deflationary othogonalization should be made to ensure that the same independent components are not estimated more than once.

5. RESULTS AND DISCUSSION

For validating the performance of the algorithm, extensive simulations were carried out on FCAICA and compared to Fast ICA, SFLOICA with two super-gaussian mixtures and two sub Gaussian mixtures.

5.1 Parameter setting

For fine-tuning this category of problem to real world applications, the parameters set are,

1. Number of initial weight vectors is eight(N=8)
2. Number of weight vectors in each group is 2 since the number of sources is 2
3. Maximum number of iteration for convergence is 100
4. The matrix A selected for this problem is given by A=[.9121 0.2292 0.4763 0.7348]

5.2 Results of supergaussian mixture

Supergaussian signals (Speech signal) have positive kurtosis value i.e greater than zero. Random variables of supergaussian signals have a spiky probability density function with heavy tails and random variables of subgaussian signals have a flat probability density function. Most of the real world signals are supergaussian in nature.The experimental results of the real world speech signals are shown in Fig.1(a) –Fig 1(f). Fig.1(a) and Fig 1(b) show two male speech signals with sampling frequency of 8 KHz.The mixtures obtained by mixing them with artificial mixing matrix A are shown in Fig 1(c) and Fig 1(d). Fig. 1(e) and Fig. 1(f) show the independent components obtained through the ICA algorithms.

The experiment was carried out with 256 samples first, for the problems of small-size. Because the defined algorithm must be capable of efficiently solving the real-world sized instances,

Journal of Theoretical and Applied Information Technology
20th May 2014. Vol. 63 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

another experiment is carried out for large-sized problem with 3000 samples each. Separation of super Gaussian mixtures are achieved at 900ns for SFLOICA, FastICA and 400ns for FCAICA.

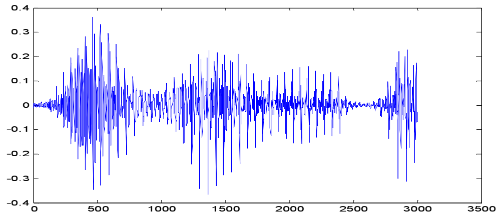

Fig 1(a). Speech signal1

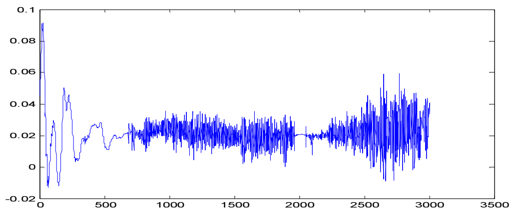

Fig.1(b) Speech signal 2

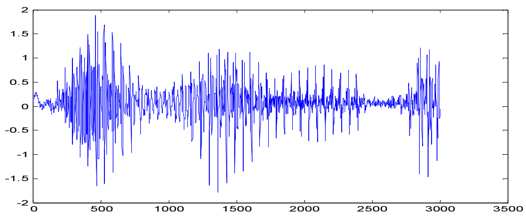

Fig 1(c) Mixture of speech signal1

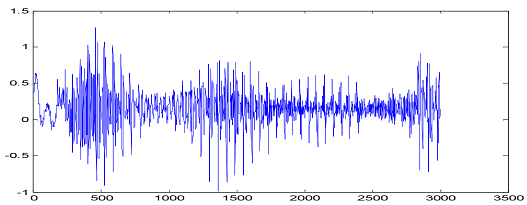

Fig 1(d) Mixture of speech signal2

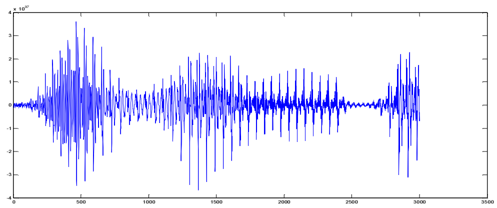

Fig 1(e) Recovered speech signal1

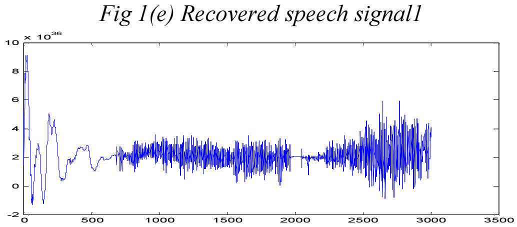

Fig 1(f).Recovered speech signal2

5.3 Convergence Analysis

The convergence analysis is done from the simulation results obtained from Modelsim 10.1 and NCsim Tool v10. Convergence speed T represents the time taken for each of the algorithms to reach convergence. The separation performance of the algorithms for subgaussian signal mixture(S1), supergaussian signal mixture (S2) and 'mixture of sub-gaussian(sawtooth) and supergaussian signals(male speech)' is shown in Fig 2.It shows that FCAICA converges faster than Fast ICA and SFLOICA due to its increased search space and adaptive nature based on fitness function

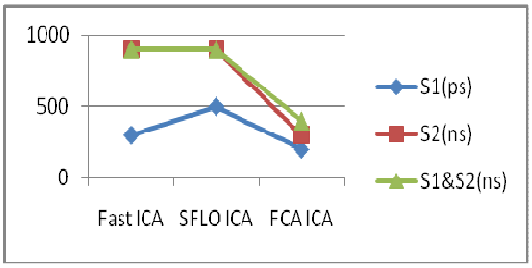

Fig 2 Convergence Time Analysis

5.4Implementation results of ICA algorithms

All the algorithms are written in very high speed integrated circuit hardware description language (VHDL) and synthesized using Quartus II Tool. The FCAICA algorithm consists of three main reconfigurable components (RCs): weight vector updating unit, convergence check unit and separation matrix calculator unit. All these modules are developed individually and integrated together to form a complete module. These RCs can be configured according to number of inputs to improve the convergence speed. The most popular Fast ICA has a unit specifically allocated for random number generation. In proposed algorithm, this random number generator is eliminated and a comparator /subtractor unit is used to decide the direction of traversal of weight vectors. This reduces the convergence time, hardware utilization and hence the power. Further investigations of ICA design implementation in FPGA for power, area and timing performance shows that Area, power and operating frequency increases in the order of FCA-ICA, Fast ICA and SFLO ICA. The results obtained from FPGA Implementation is shown in Fig.3.These results show that FCAICA has improved performance compared to other two algorithms.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

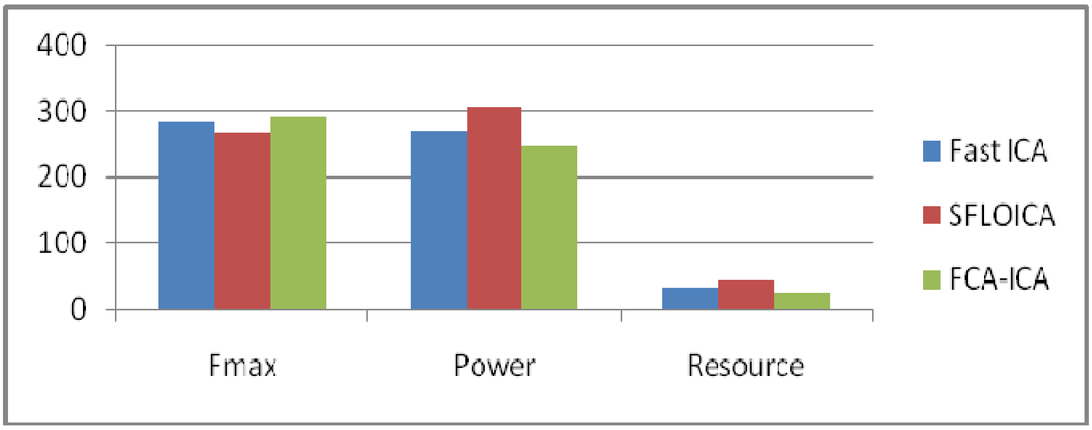

Fig 3 Comparison of Design parameters of ICA algorithms FPGA Implementation

6. CONCLUSION

In this paper, time-domain approaches to estimate the independent components from their observed supergaussian mixtures and sub-gaussian mixtures have been presented. Algorithms are synthesized in Quartus II tool .Then comparison is done with most popular Fast ICA algorithm and SFLOICA algorithm.Use of modularity and hierarchy simplifies the design, reduces the power and speeds up the convergence process of ICA. The usage of optimization algorithm enables finding optimal solution. Floating point manipulations enable increased input signal range. The oddity of the resulting system is the capability of providing faster convergence with reduced power and area. This is achieved by hardware and floating point arithmetic optimization units. The limitation of this system is that floating point ICA implementation may occupy more hardware when compared to fixed point ICA algorithms. Further research includes the application of the proposed method for other signals, such as EEG, Spread spectrum signals and images under poor SNR circumstances. Further improvement is possible by employing this technique with sources more than two.

REFERENCES

[1]. Chunsheng guo, Feng Xuan,"Moving objects detection based on kernel independent component analysis" ,*Procedia Engineering* ,Vol.15 , 2011, pp:1046 – 1050

[2]. Wei Lu, Jagath C. Rajapakse, "Eliminating indeterminacy in ICA ", *Neurocomputing* Vol. 50 , 2003 ,pp: 271 – 290

[3]. Yiu-ming Cheung, Lei Xu, "Independent component orderingin ICA time series analysis" *Neurocomputing*,Vol. 41, 2001,pp 145–152.

[4]. Hyvärinen, J. Karhunen, & E. Oja,"Independent Component Analysis", New York: *Wiley& Sons, Hoboken,* ISBN-10: 0471464198. 2004.pp: 504

[5]. Adali, T., H. Li, M. Novey and J. Cardoso, "Complex ICA using nonlinear functions",*IEEE Trans. SignalProcess*", Vol.56, 2008, pp: 4536-4544.DOI: 10.1109/TSP.2008.926104

[6]. Hyvarinen, A.,"Fast and robust fixed-point algorithms for independent component analysis", IEEE Trans. *Neural Netw.,* Vol 10, 1999,pp: 626-634. DOI: 10.1109/72.761722

[7]. Pham, D.T. and P. Garat,"Blind separation of mixture of independent sources through a quasi-maximum likelihood approach", *IEEE Trans. Signal Process*.,Vol 45, 1997,pp: 1712-1725. DOI: 10.1109/78.599941

[8]. Hyverinen, E.O., "Independent component analysis by general nonlinear Hebbian-like learning rates ", *Signal Processing*,Vol64, 1998, pp: 301-303. http://www.sciencedirect.com/science/article/pii/S0165168497001977

[9]. Herrmann, F. and A.K. Nandi, "Maximisation of squared cumulants for blind source separation" *Electronics Lett.*,Vol36, 1996,pp:1664-1665. DOI: 10.1049/el:20001135

[10]. Nandi, A.K. and V. Zarzoso, "Fourth-order cumulant based blind source separation", *IEEE Signal Process. Lett.,* Vol 3: 1996,pp:312-314. DOI: 10.1109/97.544786

[11]. Cardoso, J.F.,"Source separation using higher order moments", *Proceeding of the International Conference on Acoustics, Speech and Signal Processing,. IEEE Xplore Press,*

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Glasgow, May, 23-26, 1989pp: 2109-2112. DOI: 10.1109/ICASSP.1989.266878

[12]. Francesco, N., P. Svaizer and M. Omologo, ”Convolutive BSS of Short Mixtures by ICA Recursively Regularized Across Frequencies”, *IEEE Trans. Audio, Speech Language Process.*, Vol 19, 2011,pp:624-639.DOI: 10.1109/TASL.2010.2053027

[13]. Li, X.L. and T. Adali,”Independent component analysis by entropy bound minimization” ,*IEEE Trans. Signal Process.*,Vol58, 2010,pp:5151-5164.DOI: 10.1109/TSP.2010.2055859

[14]. Rojas, C., G. Puntonet, M. Rodriguez-Alvarez, I. Rojas and R. Martin-Clemente,”Blind Source separation in post-nonlinear mixtures using competitive learning, simulated annealing and a genetic algorithm”, *IEEE Trans. Sys. Man Cybernatics Appli. Rev.*,Vol34, 2004,pp: 407-416. DOI: 10.1109/TSMCC.2004.833297

[15]. Palaniappan, R. and C. Gupta , ”Genetic algorithm based independent component analysis to separate noise from electrocardiogram signals” ,*Proceedings of the International Conference on Engineering of Intelligent Systems, IEEE Xplore Press, Islamabad,* 2006, pp: 1-5. DOI: 10.1109/ICEIS.2006.1703159

[16]. Acharyya, A., K. Maharatna, S. Jinhong, B.M. Al-Hashimi and S.R. Gunn,”Hardware Efficient Fixed-Point VLSI Architecture for 2D Kurtotic FastICA”, *Proceedings of the European Conference on 19th IEEE European Circuit Theory and Design*, Aug 23-27, 2009, IEEE Xplore Press, Antalya, pp. 165-168. DOI: 10.1109/ECCTD.2009.5274943

[17]. Dinesh, P., N. Das and A. Routray”, Implementation of Fast-ICA: A Performance Based Comparison Between Floating Point and Fixed Point DSP Platform”, *Measurement Sci. Rev.*, Vol 11, 2011,pp:119-18.

[18]. Hongtao, D. and H. Qi,”A Reconfigurable FPGA system for parallel independent component analysis”, *EURASIP J. Embedded Syst.*,2006:1-1.DOI: 10.1155/ES/2006/23025

[19]. Celik, A., M. Stanacevic and C. Gert,”Gradient Flow Independent Component Analysis in Micropower VLSI”, 2005. http://machinelearning.wustl.edu/mlpapers/paper_files/NIPS2005_865.pdf

[20]. Jeong, Y.K. and H.J. Jang,”Adaptive parallel computation for blind source separation with systolic architecture”,*Intelli. Inf. Management*,Vol.2, 2010,pp: 46-52.DOI: 10.4236/iim.2010.21006

[21]. Hongtao, D., H. Qi and X. Wang, “Comparative Study of VLSI Solutions to Independent Component Analysis”, *IEEE Trans. Industrial Electronics,* Vol 54, 2007,pp: 548-558. DOI: 10.1109/TIE.2006.885491

[22]. Kim, C.M., HM. Park, T. Kim, Y.K. Choi and S.Y. Lee,”FPGA implementation of ICA algorithm for blind signal separation and adaptive noise cancelling*”, IEEE Trans. Neural Netw.*, Vol.14, 2003,pp: 1038-1046. PMID: 18244558

[23]. Kuo-Kai Shyu, Ming-Huan Lee, Yu-Te Wu, and Po-Lei Lee.”Implementation of Pipelined FastICA on FPGA for Real-Time Blind Source Separation” , *IEEE Trans. On Neural Networks*, Vol. 19, No. 6, 2008. PMID18541497