IPHV8I3005X
International Journal Of Advanced Research and Innovation -Vol.8, Issue .III
ISSN Online: 2319 – 9253
Print: 2319 – 9245

VLSI OUTLINE FOR CONVOLUTIVE VISUALLY IMPAIRED SOURCE DIVISION

[#1]GOLLAPALLY LAVANYA, M.Tech Student,

[#2]SD.JAKI SHARIF, Assistant Professor,

Dept of ECE,

MOTHER THERESSA COLLEGE OF ENGINEERING & TECHNOLOGY, KARIMNAGAR, TS, INDIA.

ABSTRACT: This short presents an green very-big-scale integration structure design for convolutive blind supply separation (cbss). The cbss separation community derived from the records maximization (infomax) technique is adopted. The proposed cbss chip design consists mainly of infomax filtering modules and scaling component computation modules. In an infomax filtering module, enter samples are filtered through an infomax clear out with the weights up to date with the aid of infomax-driven stochastic learning rules. As for the scaling component computation module, all operations including logistic sigmoid are incorporated and carried out via the circuit layout primarily based on a piecewise-linear approximation scheme. The proposed prototype chip is carried out thru a semi-custom design the usage of 90-nm cmos era on a die length of approximately 0.fifty four × 0.54 mm2

Keywords: *Blind Source Separation (BSS), Convolutive BSS(CBSS), Convolutive Mixing, Information Maximization (Infomax), Very-Large-Scale Integration (VLSI).*

I. INTRODUCTION

Separation of mixed sources has received extensive attention in recent years. Blind source separation (BSS)attempts to separate sources from mixed signals when most of the information for sources and mixing process is unknown. Such restrictions make BSS a challenging task for researchers. BSS has become a very important research topic in a lot of fields. Notable examples include audio signal processing, biomedical signal processing, communication systems, and image processing [1]–[3].Without a filtering effect, instantaneous mixing is considered a simple version of the mixing process of the source signals. However, for audio sources passing through an environmental filtering before arriving at the microphones, a Convolutive mixing process occurs, and Convolutive BSS (CBSS) [4] is used to recover the original audio sources. Independent component analysis (ICA) is the conventional means of solving the BSS or CBSS problem [6]. However, this method is often highly computationally intensive and introduces time-consuming processes for software implementation. More than a faster solution than software implementation, hardware solution achieves optimal parallelism [15]. Providing hardware solutions for ICA-based BSS has drawn considerable attention recently. Cohen and Andreou [7] explored the feasibility of combining above-and-sub threshold CMOS circuit techniques for implementing an analog BSS chip that integrates an analog I/O interface, weight coefficients, and adaptation blocks. This chip incorporates the use of the Herault–Jutten ICA algorithm [17]. Cho and Lee [8] implemented a fully analog CMOS chip based on information maximization (Infomax) ICA, as developed by Bell and Sejnowski [5], [19]. The chip incorporated a modular architecture to extend its use as a multichip. Apart from these analog BSS chips, various field programmable gate array (FPGA) implementations with digital architectures have been developed. Li and Lin [9] realized the Infomax BSS algorithm based on system-level FPGA design, by using Quartus II, DSP builder, and Simulink. Du and Qi [10] presented an FPGA implementation for the parallel ICA (pICA) algorithm, which focuses on reducing dimensionality in hyper spectral image analysis. The pICA algorithm consists of three temporally independent functional modules that are synthesized individually with some reconfigurable components developed for reuse. Based on Infomax BSS, Ounas et al. [11] introduced a low-cost digital architecture implemented on FPGA. This design used merely one neuron to support sequential operations of the neurons in neural network. In 2008, Shyu et al. [12] designed a pipelined architecture for FPGA implementation based on FastICA for separating mixtures of biomedical signals, including electroencephalogram (EEG), magneto encephalography (MEG), and electrocardiogram (ECG). In this design, floating-point arithmetic units were used to increase the precision of the numbers and ensure the Fast ICA performance. Although FPGA has a short development time and inexpensive verification of algorithms in hardware, its hardware architecture design is not optimized in comparison with application specific integrated circuit (ASIC) fabricated in chips. Acharyya et al. [13] designed an ASIC chip with 0.13- μm standard cell CMOS technology for 2-D Kurtotic Fast ICA.

This design is characterized by reduced and optimized arithmetic units through means of removing dividers in eigenvector computation and whitening. For portable EEG signal processing applications, Chen et al. [14] developed a low-power verylarge-scale integration (VLSI) chip fabricated using the UMC 90-nm CMOS process. However, the aforementioned ASIC chips focus on instantaneous mixing BSS. In this brief, we present a digital ASIC chip for CBSS, in which the source signals are convolutively mixed. The Convolutive mixtures are separated using the CBSS separation network [16] extended from Infomax theory. The CBSS problem was solved in the time domain mainly because in the frequency domain, the permutation and scaling ambiguity among the frequency bins must be resolved [20].

II. BSS USING INFOMAX

A. CBSS

Assume there are N source signals recorded by M sensors. This brief focuses on Convolutive mixing. The related model is mathematically expressed by

$$x_m(t) = \sum_{n=1}^{N}\sum_{k=0}^{L-1} h_{mn}(k)s_n(t-k) \quad (1)$$

Where xm, m = 1, 2, . . .,M, is a mixed signal corresponding to sensor m; sn, n = 1, 2, . . .,N, is the nth source signal; hmn is the unknown impulse response from source n to sensor m; t is the discrete time index; and L is the number of taps in convolution. CBSS, a demixing or separation process, finds separated signals that approximate the original sources. The separation process can be expressed as

$$u_n(t) = \sum_{m=1}^{M}\sum_{l=0}^{L-1} w_{nm}^{k} x_m(t-k) \quad (2)$$

Where un is the nth separated signal; wnm is the L-tap separation filter for sensor m to separated signal n; and wk nm denotes the kth tap weight of wnm. The separation process (2) can be expressed in matrix form as

$$p(s) = p(s_1, s_2, \ldots, s_N) = \prod_{n=1}^{N} p(s_n) \quad (3)$$

B. Infomax Approach for Convolutive Mixing BSS

The BSS problem assumes that statistical independence among source signals exists. Let sn denote the nth source signal. The joint probability density function of all the sources can be written as

$$u(t) = \sum_{k=0}^{L-1} \mathbf{W}^k x(t-k) \quad (4)$$

Where H denotes the differential entropy. While BSS attempts to generate separated signals close to source signals, the separation process focuses on generating output signals with zero mutual information. To minimize the mutual information, Bell and Sejnowski developed the Infomax approach [5] to learn the separating process. This approach maximizes the joint entropy of the outputs by a stochastic gradient ascent algorithm. As plain maximization of the joint entropy of the outputs may diverge to infinity [18], the Infomax approach maximizes the joint entropy of y = g(u), where g(.) refers to a nonlinear and monotonically transfer function. In Convolutive mixing, the separation process is driven by Wk, which is a matrix comprising filter components.

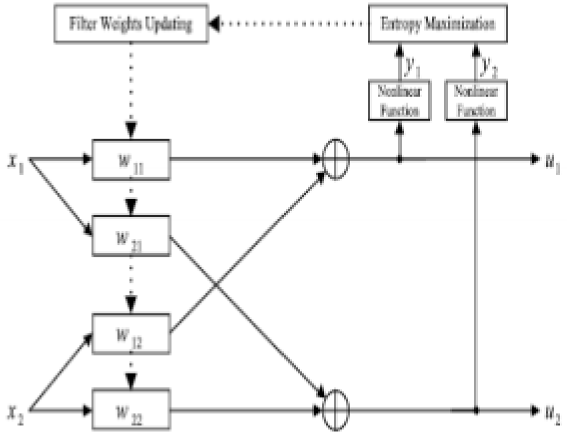

Fig 1. Infomax-based CBSS separation network for the two-source and two sensor case.

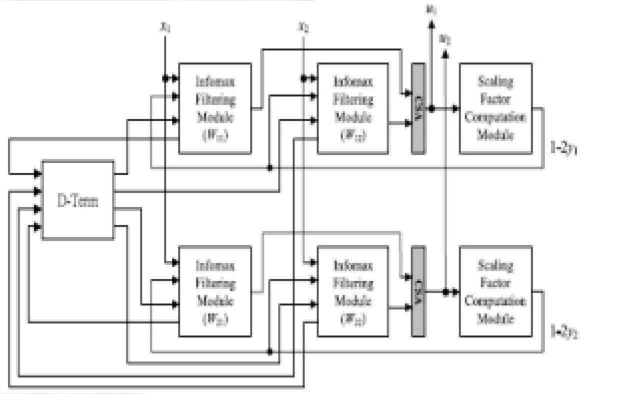
x_1
x_2
u_1
u_2
Infomax Filtering Module (W_{11})
Infomax Filtering Module (W_{12})
CSA
Scaling Factor Computation Module
$1-2y_1$
D-Term
Infomax Filtering Module (W_{21})
Infomax Filtering Module (W_{22})
CSA
Scaling Factor Computation Module
$1-2y_2$

Fig 2. Block diagram of the proposed CBSS chip that contains four Infomax filtering modules, two scaling factor computation modules, and a D-term unit.Two CSAs are used to sum up the Infomax filtering outputs.

III. PROPOSED VLSI BLIND SOURCE SEPARATOR

Fig. 2 shows the block diagram of the proposed CBSS chip. The CBSS chip consists mainly of two functional cores: Infomax filtering module and scaling factor computation module. Additionally, the Infomax filtering outputs are summed up using two small carry-save adders (CSAs). The current prototype chip is used for two sources and two sensors by adopting four Infomax filtering modules and two scaling factor computation modules.

A. VLSI Architecture for Infomax Filtering Module Fig. 1 depicts the CBSS separation network, which contains four causal FIR filters. These filters are adaptive because their tap coefficients are altered by stochastic learning rules derived from the Infomax approach and are thus referred to herein as the Infomax adaptive filter or the Infomax filter. Equations (8) and (9) describe the stochastic learning rules to adjust the Infomax filter weights. In the Infomax filtering module, an input sample passes through lower and upper register chains. The input samples passing through the lower and upper register chains are multiplied with filter weights and scaling factors, respectively.

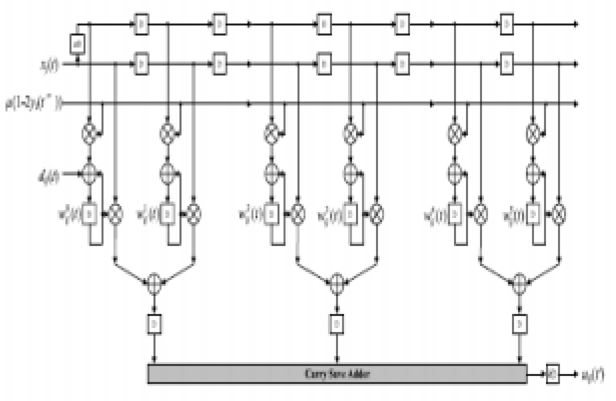

Fig 3. Example of the proposed Infomax filtering module. The multiplication results of all of the taps are accumulated by a two-stage summation.

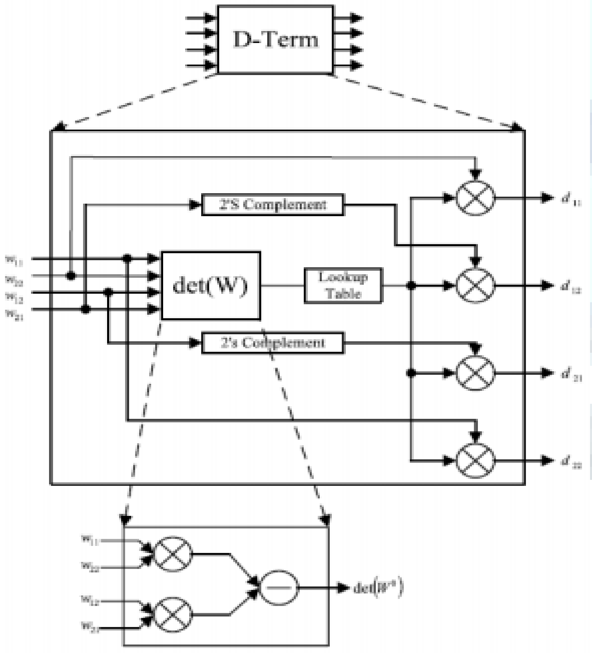

Fig 4. Architecture of the D-term unit, which comprises a determinant circuit to obtain det W 0 and a lookup table to generate the inverse of det W 0

The multiplication results of all of the taps are accumulated by a two-stage summation. The first stage adopts carry look a head adders to generate the intermediate addition results for multiplication of every two successive taps. The second stage sums the above intermediate addition results by using a carry save addition scheme. A CSA can accept more than two data inputs. As this CSA may accept many intermediate addition results, reducing the critical path as low as 1Ta + 1Tm can be achieved by partitioning this CSA with pipeline registers. Here,Ta and Tm denote the critical paths of the carry lookahead adder and multiplier, respectively. In Fig. 3, k pipeline registers are assumed to partition the CSA.

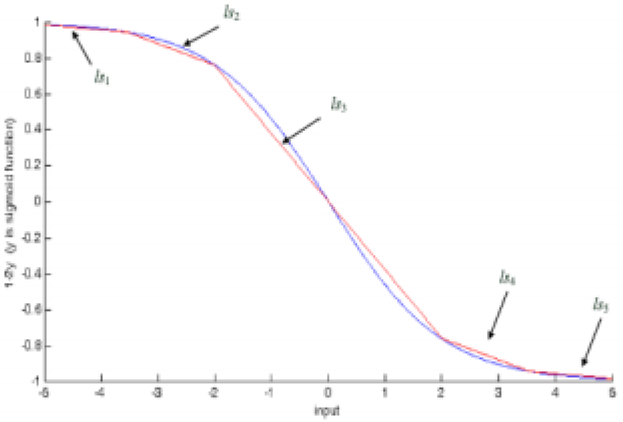

Fig 5. Five line-segment approximation to the scaling factor computation, where lsi denotes the ith line segment.

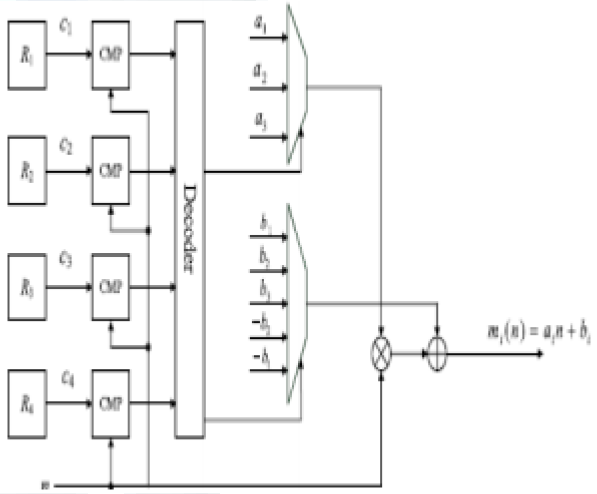

Fig 6. Proposed scaling factor computation module. Four comparators and a decoder are used to determine the correct line segment.

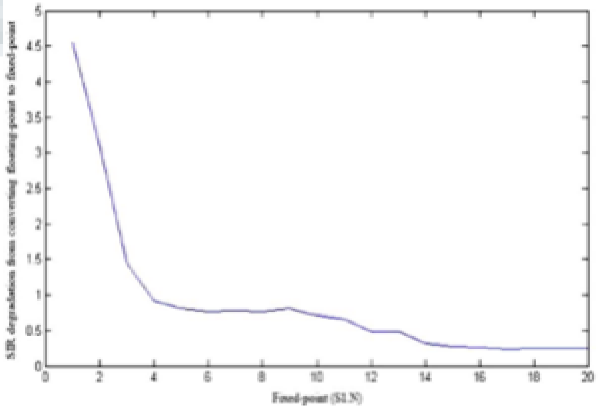

Fig 7. SIR Degradation from converting the floating point algorithm into the fixed-point algorithm.

The scaling factor used in filter weight updating, as described in (8) and (9), is obtained by calculating s(t) = 1 − 2y(t),where y(t) = (1+e−u(t))−1. For a straightforward computation flow, once y(t) is available, −2y(t) can be generated first using 2's complement and a left shift to y(t).The scaling factor s(t) is then obtained by summing up

−2y(t) and one. The above procedure is simple. The emphasis of architecture design in the scaling factor computation module should thus lie in the logistic sigmoid computation. In this brief, the logistic sigmoid computation is achieved based on a linear piecewise scheme [23], [24]. The scaling factor commutation is approximated directly rather than performing logistic sigmoid computation first and then calculating 1 − 2y(t). In our numerical analysis, five line segments are sufficient to approximate (11) with a negligible error. Let lsi, i = 1, 2, . . . , 5 denote the ith line segment, and ci represent the connected point between two consecutive line segments, lsi and lsi+1. Fig. 5 plots the approximation results of the five line segments.The average error between the approximate value and the floating value obtained using (11) is around 2.88%. To implement the line-segment approximation, the circuit design for scaling factor computation is to calculate single variable linear equations. Assume that the equation of lsi is mi(n) = ain + bi, i = 1, 2, . . . , 5, where n = ui(t). Fig. 6 describes the proposed circuit for the scaling factor computation module. The linear equation evaluation with input ui(t) and equation parameters ai and bi are implemented using a multiplier and an adder. A line segment is selected by two multiplexers to choose corresponding ai and bi. As the slopes of ls1 and ls5 are the same, these two line segments share the equation parameters a1. In the same manner, line segments ls2 and ls4 share the equation parameters a2. Furthermore, according to the symmetry in Fig. 8, the bias used for line segment ls5, e.g., −b1, is the negative of the bias b1 used for line segment ls1. In addition, line segments ls4 and ls2 use biases −b2 and b2, respectively.

IV. RESULTS

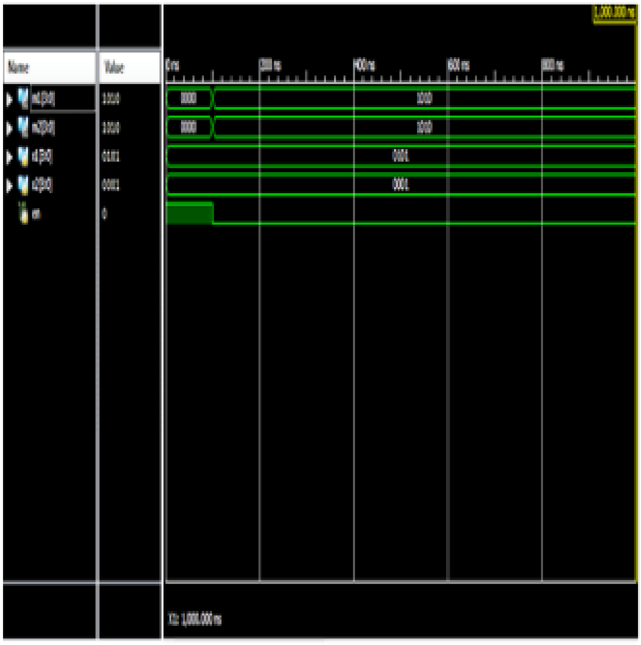

Fig 8. Simulation result for the proposed system.

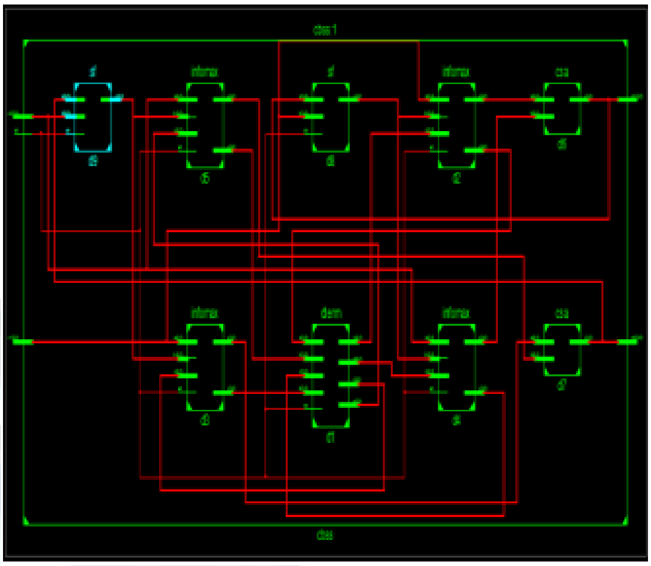

Fig 9. RTL schematic view for the proposed system.

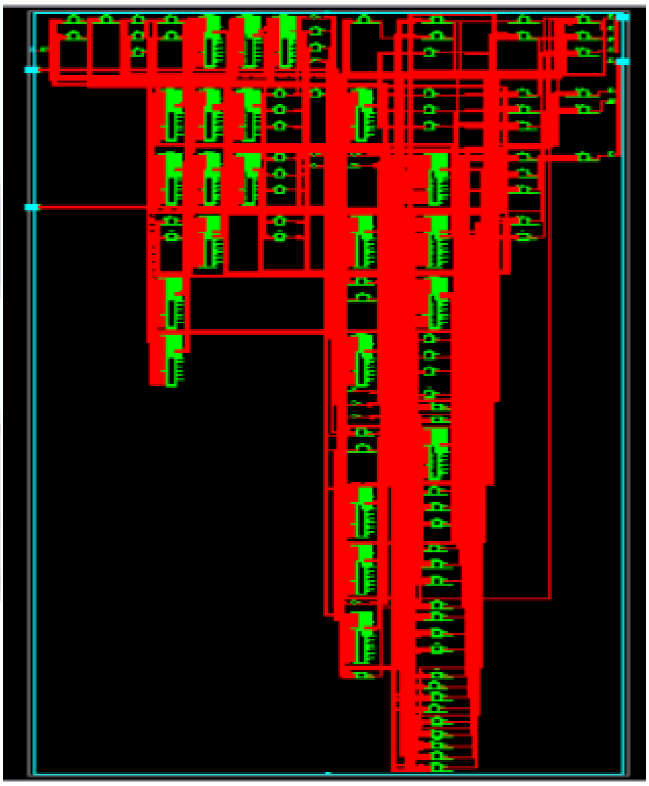

Fig 10. Technology schematic view for the proposed system.

V. CONCLUSION

In this brief, an efficient VLSI architecture design for CBSS has been presented. The architecture mainly comprising Infomax filtering modules and scaling factor computation modules performs CBSS separation network derived from the Infomax approach. With TSMC 90-nm CMOS technology, the die size of the proposed ASIC chip is roughly 0.54 × 0.54 mm2. For the 1.8-V power supply, the maximum clock rate is 100 MHz. The power dissipation is roughly 54.86mWunder the 100-MHz clock rate. The proposed CBSS ASIC chip can be used in preprocessing and integrated with other audio processing chips and peripheral components to form a whole audio processing System.

REFERENCES

[1] G. Zhou, Z. Yang, S. Xie, and J. M. Yang, "Online blind source separation using incremental nonnegative matrix factorization with volume constraint," IEEE Trans. Neural Netw., vol. 22, no. 4, pp. 550–560,Apr. 2011.

[2] M. Li, Y. Liu, G. Feng, Z. Zhou, and D. Hu, "OI and fMRI signal separation using both temporal and spatial autocorrelations," IEEE Trans. Biomed. Eng., vol. 57, no. 8, pp. 1917–1926, Aug. 2010.

[3] A. Tonazzini, I. Gerace, and F. Martinelli, "Multichannel blind separation and deconvolution of images for document analysis," IEEE Trans. Image Process., vol. 19, no. 4, pp. 912–925, Apr. 2010.

[4] H. L. N. Thi and C. Jutte, "Blind source separation for convolutive mixtures," Signal Process., vol. 45, no. 2, pp. 209–229, Aug. 1995.

[5] A. J. Bell and T. J. Sejnowski, "Blind separation and blind deconvolution: An information-theoretic approach," in Proc. Int. Conf. Acoust., Speech, Signal Process., May 1995, vol. 5, pp. 3415–3418.

[6] A. Hyvärinen and E. Oja, "Independent component analysis: Algorithms and applications," Neural Netw., vol. 13, no. 4/5, pp. 411–430,May/Jun. 2000.

[7] M. H. Cohen and A. G. Andreou, "Analog CMOS integration and experimentation with an autoadaptive independent component analyzer," IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process., vol. 42, no. 2, pp. 65– 77, Feb. 1995.

[8] K. S. Cho and S. Y. Lee, "Implementation of InfoMax ICA algorithm with analog CMOS circuits," in Proc. Int. Workshop Independent Compon. Anal. Blind Signal Separation, Dec. 2001, pp. 70–73.

[9] Z. Li and Q. Lin, "FPGA implementation of Infomax BSS algorithm with fixed-point number representation," in Proc. Int. Conf. Neural Netw. Brain, 2005, vol. 2, pp. 889–892.

[10] H. Du and H. Qi, "An FPGA implementation of parallel ICA for dimensionality reduction in hyperspectral images," in Proc. IEEE Int. Geosci.Remote Sens. Symp., Sep. 2004, pp. 3257–3260.

[11] M. Ounas, R. Touhami, and M. C. E. Yagoub, "Low cost architecture of digital circuit for FPGA implementation based ICA training algorithm of blind signal separation," in Proc. Int. Symp. Signals, Syst. Electron., 2007,pp. 135–138.

[12] K. K. Shyu, M. H. Lee, Y. T. Wu, and P. L. Lee, "Implementation of pipelined FastICA on FPGA for realtime blind source separation," IEEE Trans. Neural Netw., vol. 19, no. 6, pp. 958–970, Jun. 2008.

[13] A. Acharyya, K. Maharatna, J. Sun, B. M. Al-Hashimi, and S. R. Gunn, "Hardware efficient fixed-point VLSI architecture for 2D Kurtotic FastICA," in Proc. Eur. Conf. Circuit Theory Des., Aug. 23–27, 2009, pp. 165–168.

[14] C. K. Chen et al., "A low power independent component analysis processor in 90 nm CMOS technology for portable EEG signal processing systems," in Proc. IEEE Int. Symp. Circuits Syst., May 15–18, 2011, pp. 801–804.

[15] H. Qi and X. Wang, "Comparative study of VLSI solutions to independent component analysis," IEEE Trans. Ind. Electron., vol. 54, no. 1,pp. 548–558, Feb. 2007.