EMBEDDED SYSTEM CHIP DESIGN FOR MEDICAL IMAGE INDEXING AND RECOGNITION

Long Xie*, **Youichi Kigawa***, **Masatoshi Ogawa***
Harutoshi Ogai*, **Hirokazu Abe#**

**Department of information, production and system,*
Waseda University, Japan
#Kushu Measurement & Control Inc.
xielong@akane.waseda.jp, kigawa@ akane.waseda.jp,
stream@asagi.waseda.jp, ogai@waseda.jp
abe@qk_net.co.jp

Abstract: An embedded system chip is developed for medical image indexing and recognizing through huge medical image database. Feature points which can represent the image are extracted using ICA algorithm after the image is preprocessed and different recognition methods are compared to get the optimized one. The entire project of hardware and software co-design is implemented on Xilinx FPFA with the consideration of high efficiency, low power consuming and easy integration with other devices.

Keyword: Medical, Image, Recognition Independent Component Analysis, SoC

1. INTRODUCTION

With the development of computer and electronic technology, more and more medical images from hospitals, medical universities and libraries are stored and processed in digital format which have different characteristic (Vittorio and Lawarence, 2002) to common images. The embedded system chip is developed with the thinking of system on chip (SoC) to help doctors and researches to get the desirable images as soon as possible from image database and to make medical expert system get the diagnosis and treat results efficiently and correctly. Also it is necessary that the chip is low consuming and can cooperate with other parts of medical equipment easily. The structure of this paper is show as below. In Section2, the independent component analysis (ICA) algorithm based on the measurement of nogentropy is briefly introduced and this algorithm is used to extract feature points from the original images. Following in Section 3, all the algorithms are programmed on Matlab and C++ platforms, the difference between recognition methods are compared according to the extracted feature points, also the preprocessing and division ways of images are tested to improve successful recognition rate. The method of developing the hardware platform and the procedure of hardware and software co-design on Xilinx FPGA to get the final chip is introduced in Section 4. Furthermore the example data is tested to verify stability and efficiency of the entire system cooperating with image database.

2. FIX-POINT ALGORITHM OF INDEPENDENT COMPONENT ANALYSIS

ICA (Comon, 1994, Lee, *et al.* 2000) is a statistical and computational technique for revealing hidden factors that underlie in sets of random variables, measurements, or signals. Assume that n mixed signals are denoted by column vector X whose elements are demoted by $x_1, x_2, ..., x_n$ and the independent components of column vector are defined as S with elements $s_1, s_2, ..., s_n$. Also define A with elements a_{ij} as mixing matrix. The above model is written as:

$$X = AS \tag{1}$$

ICA is originally proposed to solve the blind source separation problem, to recover n original source signals S from the linearly mixed signal X, while assuming as little as possible about the natures of mixing matrix A. By defining the matrix $W = A^{-1}$, convert the model according to (1) as:

$$S = WX \tag{2}$$

So that the independent components can be calculated from the linear transformation by W.

PCA (principal component analysis) (Ruymagaart and Robust, 1981, Xu and Yuille, 1995) is a preprocessing step to ICA which reduces the redundancy of mixed signals on a certain level. In the PCA transformation the vector X is first centered by subtracting its mean then $E\{X\} = 0$. Next find a rotated orthogonal coordinate system such that the elements of X in the new coordinates become uncorrelated and X is linearly transformed to another vector Y with m elements $m < n$, so that the redundancy induced by the correlation is removed. In the PCA by variance maximization define that

$$C_x = E\{XX^T\} \tag{3}$$

is the $n \times n$ covariance matrix of X and $e_1, e_2, ... e_n$ are the unit-length eigenvectors of the matrix. The eigenvectors are ordered according to the ordering eigenvalues $d_1, d_2, ... d_n$ satisfying $d_1 > d_2, ... d_n$. Then m dimensional vector Y is:

$$Y = EX \tag{4}$$

got by choosing the first m eigenvectors of $e_i (i = 1...m)$ as the row vector of E, refer (Tipping and Bishop, 1999) for details.

After reduce the redundancy of mixed signals, the ICA problem can be greatly simplified if the signals are first whitened (Cardoso and Laheld, 1996) with linear transformation of

$$Z = VY \tag{5}$$

Define $V = D^{-1/2} E^T$, where $E = (e_1, e_2, ... e_m)$ whose columns are the m eigenvectors of C_y ,which has the same mean as in (3) and $D = diag(d_1, d_2, ... d_m)$ is the diagonal matrix of the eigenvalues of C_y. Recalling that C_y can be written as $C_y = EDE^T$, with E an orthogonal matrix satisfying $E^T E = EE^T = I$, it holds:

$$E\{ZZ^T\} = VE\{XX^T\}V^T = D^{-1/2} E^T EDE^T ED^{-1/2} = I \tag{6}$$

then it is demonstrated that Y is whitened by V.

The fixed-point algorithm (Hyvärinen, 1999, Hyvärinen and Oja, 2000) is realized using C++ and Matlab to find the independent components of whitened vector Z. Suppose a transformation as (2) is defined as:

$$S = WZ \tag{7}$$

where $S = \{s_1, s_2, .. s_m\}$ is the vector of independent component. Define the mutual information

$$I(s_1, s_2, .. s_m) = J(S) - \sum_i J(S_i) \tag{8}$$

of s_i which is a measurement of the independence of random variables, $J(S)$ is negentropy given by $J(S) = H(S_{gauss}) - H(S)$, S_{gauss} is a Caussian random variable of the same covariance matrix as S, and H is the differential entropy (Papoulis, 1991). To get the desirable matrix W from maximizing the mutual information (8) of independent component s_i, maximize

$$J(s_i) = [E\{G(W^T Z)\} - E\{G(v)\}]^2 \tag{9}$$

under the constrain of $E\{G(W^T Z)^2 = 1\}$ with appropriate function G. The final result of W is got by Newton iteration of

$$W^+ = W - \mu[E\{Zg(W^T Z)\} - \beta W]/[E\{g'(W^T Z)\} - \beta]$$
$$W = W^+ / \|W^+\| \tag{10}$$

where β can be approximated as $E\{Z^T Zg(W^T Z)\}$, and μ is a step size parameter that make the solution converge faster and g is derivative function of G defined in (9), which decide the converge speed of ICA algorithm (Hyvärinen, Hoyer and Inki, 2001). For several units of mixed signals, the out put matrix $W_M = (W_1 Z, W_2 Z, ... W_t Z)$ must be decorrelated after every iteration to prevent the units from converging to the same maxima. Follow the iterative algorithm:

$$W_M = W_M \sqrt{\|W_M C_z W_M{}^T\|}$$

$$W_M = \frac{2}{3} W_M - \frac{1}{2} W_M C_z W_M{}^T W_M \tag{11}$$

Then the matrix according to(10) is symmetrically decorrelated, see (Attias, 1999).

3. IMAGE RECOGNITION AND COMPUTER SIMULATION OF ALGORITHM

Primary part functions of hardware system are first simulated on PC before implement to FPGA to test

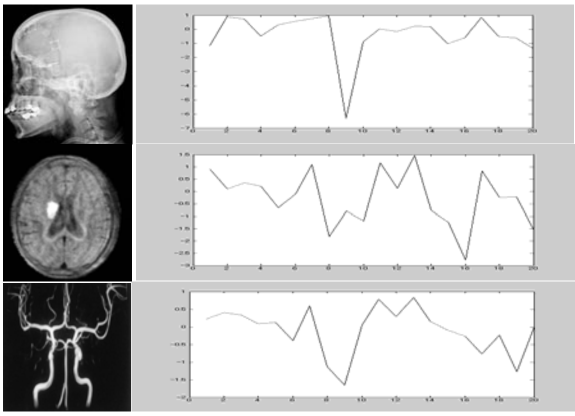

Fig.1. Feature points extracted from images

the precision and relative execution time which are important for embedded system. According to appropriate precision and time consuming one image of format JPG is divided into $1,4,\ldots,n^2$ with integer n, partitions and digitize each partition as a column of mixed matrix X_M. The feature points of image are extracted using ICA algorithm introduced above. Then they are compared with the feature points of images stored in the database which are also divided into same partitions and processed by ICA algorithm. Parts of the feature points extracted from some images are shown in Fig.1.

A proposed similarity calculation method is developed to recognize the desirable image from the image database according to the character of this project and it is compared with the common minimum mean-square error method (Jain, *et al*. 2000), the sketch map is shown in Fig.2.

Step1: Get the value of independent n and set the appropriate number M of feature points of original key images which is compromise of precision and execution time.

Step2: Record the position and sign of M feature points of the original image from the side of maximum absolute value of these points.

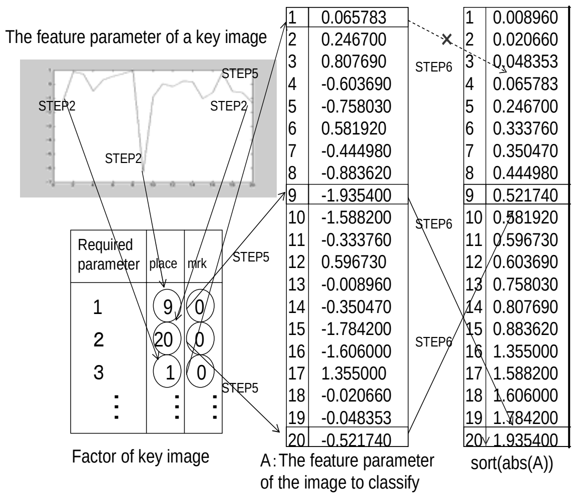

Fig.2. Explanation of similarity calculation of image

Table 1 Comparison of the two recognition methods

Calculation of similarity	Number of division	Rate of recognition(%)
minimum mean-square error method	1	34
	4	50
	9	46
Proposed recognition method	1	22
	4	41
	9	52
	16	57

Step3: Find the most similar point from the feature points of an image in database which means that difference of the absolute value of the two points is the smallest.

Step4: Calculate the similarity according to

$$similarity_d = \prod_{m=0}^{M-1} \frac{\mu x}{n-m} \qquad (12)$$

x is feature point position of the image in database, $\mu=1$ if the signs of two points are same and $\mu=0$ if opposite.

Step5: Process the next feature point of original image and calculate the similarity rate if M is not satisfied.

Step6: If the image is divided to number D, the similarity is accumulated following the results of (12) as:

$$Similarit\,y' = \sum_{d=1}^{D} similarity_d \qquad (13)$$

The calculation time is reduced and precision is improved using the method above comparing with minimum mean-square error method, usually an image is divided into 9 parts according to the requirement of real-time processing. Table 1 shows the comparing result of successfully recognizing the desirable image from image database between the two methods and the two operations are down on the same software conditions.

Also the influence of division number of an image to the successful recognition rate is tested and recommended in the software environment before implement to hardware system. A set of calculation efficiency is also tested according to the division number. Fig.3 shows the relationship of different division number, feature points and recognition rate. This result is the average value for sets of medical images of different kind processed on same software environment. Before extracting the feature points using ICA method, the image can be preprocessed to

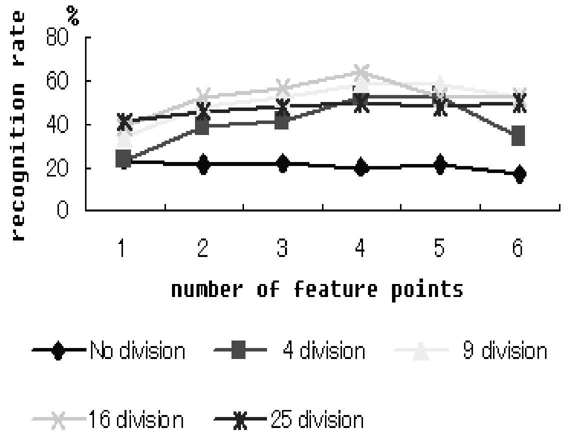

Fig.3. Relationship of different division number feature points and recognition rate

get the most necessary section and remove noise. The difference before and after preprocessing of effective feature points is shown as Fig.4, where the image is processed from vertical and horizontal directions to get more precise feature points.

The number of effective feature points is increased and the recognition rate is also improved. Table 2 gives the results of recognition rate with preprocessing using proposed recognition method under different division number.

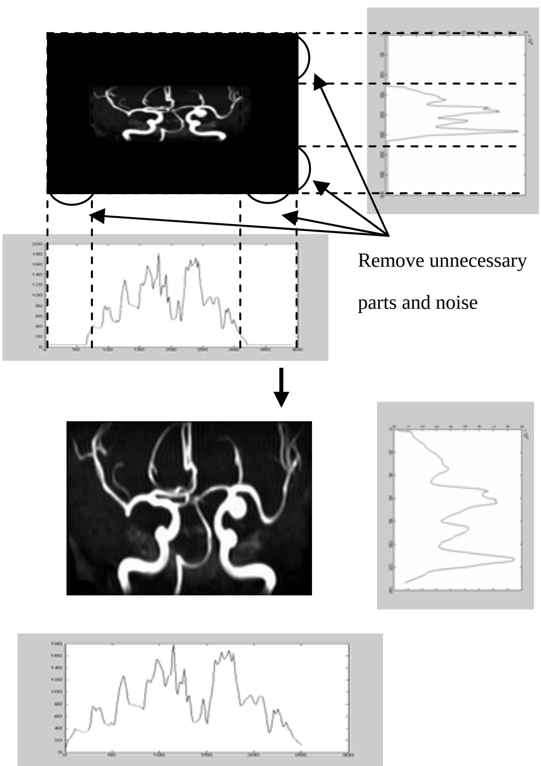

Fig.4 Comparing the feature points difference before and after preprocessing

Table 2 Result of recognition rate with preprocessing

Number of division	Number of Feature points	Rate of recognition(%)
9	4	68
	5	72
16	4	59
	5	68
	6	36

4. BUILD THE HARDWARE AND SOFTWARE ON FPGA

Verifying the design idea on FPGA is a good way before the final production of ASIC, or FPGA can be used directly when its price is acceptable. The entire system functions are implemented on Memec design development board Virtex2 P7-672 with P160 communications module. This board uses Xilinx FPGA supporting IBM PowerPC (Xilinx Inc. 2003a), also the communications module provides data interface to exchange data with outside.

After simulate the main functions of software system on the Matlab platform, C++ source code of this project is developed and compiled using GNU compile to verify the stability and test the relative time consuming on PC, also PowerPC supports C++ language and GNU gcc compiler so that the C++ source code can be run on PowerPC directly. Xlinx company provide a serial of development tools which are efficient to build the hardware and software systems. The frame of hardware and software co-design is as Fig.5, which shows the development of hardware platform of IP cores and software platform of C++ source code with Xilinx EDK tools (Xilinx Inc. 2003b).

The hardware system is developed using a serial of IP cores and basing on bus structure. A PowerPC is connected to two 16K PLB memories over the Processor Local Bus, which are used to store the data

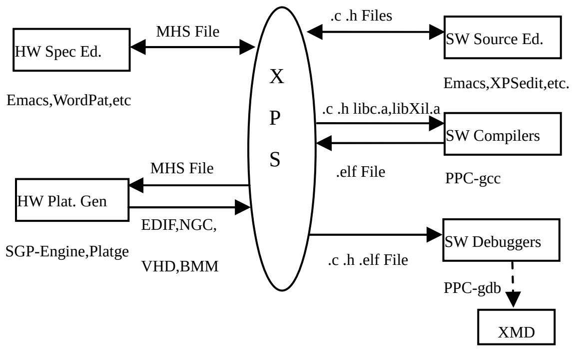

Fig.5. Hardware and software co-design platform

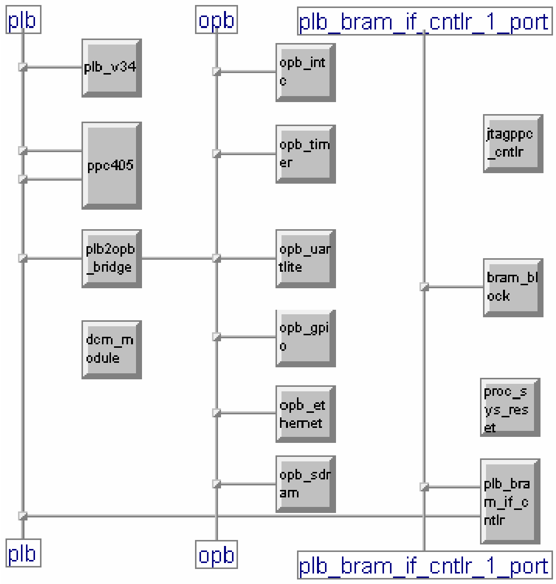

Fig.6. Hardware structure on FPGA

and instructions that used by CPU frequently because of higher processing speed. A UARTLite and an Ethernet 10/100 MAC are connected to On-chip Peripheral Bus(OPB) providing the interface of data transfer between this chip and outside device, such as other parts of equipment and image database. A 32M OPB memory is connected to the OPB bus storing the remaining data and instructions of compiled source code. This system also contains an interrupt controller to expand the number of interrupt inputs to PowerPC and request service such as data transfer and error processing. A timer serves to coordinate and handle the event of time expiring from UARTLite and MAC to complete the required system functions. A PLB to OPB bridge is used to connect the two different bus interface. The hardware structure is shown as Fig.6, refer to (Xilinx Inc. 2003c) for details. The hardware of IP cores is synthesized and verified by EDA tool and also the source code and the libraries for memory management and interrupt control are compiled and linked to construct the software system. Finally the hardware part and software part are downloaded to FPGA with download.bit file.

5. CONCLUSION

Algorithm analysis, software simulation and hardware implement are introduced in this paper. ICA algorithm is used to extract the feature points of images making the image recognition possible and efficient The algorithm is tested and verify on PC with different recognition methods and image division ways. The embedded system FPGA is developed and tested with some example data. This design also can be used in the fields of security and fault detection system with some modification. Next step is to improve the efficiency of ICA algorithm and make it run faster; also other recognition algorithms in the field of pattern recognition will be tried and the successful recognition rate may be improved. Furthermore the parts of outside interface circuit should be integrated into FPGA possibly to make the entire system more powerful and efficient.

REFERENCE

Attias, H. (1999). Independent factor analysis. *Neural Computation*, **vol. 11**, pp. 803–851.

Cardoso, J. F. and Laheld, B. H. (1996). Equivariant adaptive source separation. *IEEE Transactions on Signal Processing*, **vol.44**, pp. 3017-3030.

Comon , P. (1994). Independent component analysis, a new concept?. *Signal Process*, **vol. 36**, pp. 287–314.

Hyvärinen, A. (1999). Fast and Robust Fixed-Point Algorithms for Independent Component Analysis. *IEEE transaction on neural networks*, **vol. 10**, pp. 626-634.

Hyvärinen, A. and Oja, E. (2000). Independent Component Analysis: Algorithms and Applications. *Neural Networks*, **vol.13**, pp. 411-430.

Hyvärinen, A, Hoyer, P. O. and Inki, M (2001). Topographic independent component analysis. *Neural Computation*, **vol. 13**, pp. 1527–1558.

Jain, A. K., Duin, R. P. W. and Mao, J. (2000). Statistical Pattern Recognition: A Review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **vol. 22**, pp. 4–37.

Lee, T. W., Lewicki, M. S. and Sejnowski, T. J. (2000). ICA mixture models for unsupervised classification of non-gaussian sources and automatic context switching in blind signal separation. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, **vol. 22**, pp. 1–12.

Papoulis, A. (1991). *Probability, Random Variables, and Stochastic Processes, 3rd*. McGraw-Hill, USA.

Ruymagaart, F. and Robust, A. (1981). Principal Component Analysis. *J.Multivariate Anal*, **vol. 11**, pp. 485–497.

Xu, L. and Yuille, A. (1995). Robust principal component analysis by self-organizing rules based on statistical physics approach. *IEEE Transactions on Neural Networks*, **vol. 6**, pp.131–143.

Tipping, M. and Bishop, C. (1999). Probabilistic principal component analysis. *Journal of the Royal Statistical Society B*, **Vol.61**, pp. 611-622.

Vittorio, C. and Lawrence, D.B. (2002), *Image Databases*, chapter 4. John Wiley & Sons Inc.,USA.

Xilinx Inc. (2003a). *Embedded System Tools Guide. Xilinx*, USA.

Xilinx Inc. (2003b). *Xilinx Device Drivers documentation*. Xilinx, USA.

Xilinx Inc. (2003c). *PowerPC Processor Reference Guide*. Xilinx, USA