FastICA Algorithm for the Separation of Mixed Images

ARTI KHAPARDE* M.MADHAVILATHA+
M.B.L.MANASA* P.ANIL BABU* S.PRADEEP KUMAR*
Department of Electronics and Communication Engineering
*Aurora's Technological And Research Institute,
+ Jawaharlal Nehru Technological University,
Hyderabad,
INDIA.
Email: artikhaparde@gmail.com

Abstract: - Independent component analysis is a generative model for observed multivariate data, which are assumed to be mixtures of some unknown latent variables. It is a statistical and computational technique for revealing hidden factors that underlies set of random variable measurements of signals. A common problem faced in the disciplines such as statistics, data analysis, signal processing and neural network is finding a suitable representation of multivariate data. The objective of ICA is to represent a set of multidimensional measurement vectors in a basis where the components are statistically independent. In the present paper we deal with a set of images that are mixed randomly. We apply the principle of uncorrelatedness and minimum entropy to find ICA. The original images are then retrieved using fixed point algorithm known as FastICA algorithm and compared with the original images with the help of estimated error. The outputs from the intermediate steps of algorithm such as PCA, Whitening matrix, Convergence of algorithm and dewhitening matrix are also discussed.

Keywords: - PCA, ICA, Statistical independence, Non-gaussianity, Maximum Likelihood, Feature Extraction.

1 Introduction

ICA is a method for finding underlying factors or components from multivariate data. The approach that distinguishes ICA from other methods is that it looks for components that are both statistically independent and non-Gaussian.

In reality, the data often does not follow a gaussian distribution and the situation is not as simple as those methods of factor analysis, projection pursuit or PCA assumes. Many real world data sets have super Gaussian distributions. Hence the probability density of the data is peaked at zero and has many tails, when compared to a Gaussian density of the same variance. This is the starting point of ICA where we try to find statistically independent components in the general case where the data is non gaussian .In this paper we provide the different estimation principles of ICA and their algorithms.

This emerging technique appears as a powerful generic tool for data analysis and the processing of the multi sensor data recording. In ICA, "independence" should be understood in its strong statistical sense: it is not reduced to decorrelation; because for the purpose of ICA. Second order statistics fail to capture important features of a data set, as there are many linear transforms which decorrelate the entries of a random vector. [1].

In this paper we try to analyze the results for FASTICA algorithm [1], applied to the mixed data of 4 images, for different non-linearities. The images are mixed in the following way:

$$M = rnd(s_1)*I_1 + rnd(s_2)*I_2 + rnd(s_3)*I_3 + rnd(s_4)*I_4 \quad \ldots(1)$$

Where **S**n is the random vector of size of image. Thus we are taken the more generalized condition as compared to the condition taken in [2] where **S**n is assumed to be a real integer

This paper is organized as follows. In section 2, we try to explain the ICA and its general assumption for indentifiability of its model. In section 3, we consider the different contrast function for maximization of non guassianity such as kurtosis and negentropy. Section 4 deals with the algorithm that we apply to mixed image to find ICA. Section 5 deals with the result and conclusion.

2 Independent Component Analysis

2.1 Definitions of ICA

In the literature there are three different basic definition of ICA [1] Here we are using the basic definition that, ICA of the random vector X consists of finding a linear transform

$$X = AS \quad \text{----------- (2)}$$

so that the components si are as independent as possible, w.r.t. some maximum function that measures independence. This definition is known as a general definition where no-assumptions on the data are made [1].

Independent component analysis (ICA) is the decomposition of a random vector in linear components which are "as independent as possible". Here, 'independence' should be understood in its strong statistical sense: it goes beyond second order decorrelation and thus involves the non-gaussianity of the data. The ideal measure of independence is the higher order cumulants like kurtosis and mutual information and is known to be related to the entropy of the components. Taking this into consideration there has been considerable amount of research on the algorithms for performing ICA [1]-[7].

2.2 Indentifiability of the Model

In addition to the basic assumption of statistically independence, by imposing the following fundamental restrictions, the indentifiability of the noise free ICA model can be assured.

1. All the independent components S_i, with the possible exemption of one component, must be non-Gaussian
2. The number of observed linear mixtures m must be at least as large as the number of independent components n; i.e. m> p
3. The matrix A must be of full column rank.

3 ICA Algorithm

3.1 ICA By Maximization Of Non Gaussianity

One of the simple & intuitive principles for estimating the model of ICA is based on maximization of non-gaussianity. Non Gaussian components are Independent [1].Central limit theorem states that the distribution of a sum of independent random variables tends towards a gaussian distribution, under certain conditions. Estimating the independent components can be accomplished by finding the right linear combinations of the mixture variables, since we can invert the mixing as

$$\mathbf{S} = \mathbf{A}^{-1}\mathbf{X}. \qquad \ldots\ldots(3)$$

Thus to estimate one of the independent components, we can consider a linear combination of x_i Let us denote this by

$$\mathbf{Y} = b^T\mathbf{X} = b^T\mathbf{AS}. \qquad \ldots\ldots. (4)$$

Hence if b were one of the rows of $\mathbf{A}^{-1}$, this linear combination $b^T\mathbf{X}$ would actually equal one of the independent components.

But in practice we cannot determine such 'b' exactly because we have no knowledge of matrix **A**, but we can find an estimator that gives a good approximation. In practice there are two different measures of Non- Guassianity

3.1.1Kurtosis

The classical measure of non-gaussianity is kurtosis or the fourth order cumulant. It is defined by

$$\mathrm{Kurt}(y) = E\{y^4\} - 3\,(E\{y^2\})^2 \qquad \ldots..(5)$$

As the variable y is assumed to be standardized we can say

$$\mathrm{Kurt}(y) = E\{y^4\} - 3 \qquad \ldots\ldots..(6)$$

Hence the kurtosis is simply a normalized version of the fourth moment $E\{y^4\}$. For the gaussian case the fourth moment is equal to 3 and hence kurt (y) =0. Thus for gaussian variable kurtosis is zero but for non-gaussian random variable it is non-zero.

3.1.2 Negentropy

Negentropy is another very important measure of non-guassianity. To obtain a measure of non-gaussianity that is zero for a gaussian variable and always non negative for a non Gaussian random variable, we can use a slightly modified version of the definition of differential entropy called negentropy.

Negentropy J is defined as

$$J(y) = H(y_{gauss}) - H(y) \qquad \ldots. (7)$$

Where y_{gauss} is a gaussian random variable of the same covariance matrix as y.

3.2 Negentropy in terms of Kurtosis

As the gaussian variable has the largest entropy among all the random variables, the negentropy for the random variables will always be positive and it is zero if and only if it is a gaussian variable. Moreover, the negentropy has an additional property that it is invariant for invertible transformation.

But the estimation of negentropy is difficult, as it would require an estimate of the pdf. Therefore in practice negentropy is approximated by using higher order moments.

$$J(y) \approx \frac{1}{12} E\{y^3\}^2 + \frac{1}{48} \mathrm{kurt}(y)^2 \ldots (8)$$

Again the random variable y is assumed to be standardized

In order to increase the robustness another approach is to generalize the higher order cumulant approximation. So that it uses expectations of general non-quadratic functions. As a simple case, we can take any two non-quadratic functions G_1 & G_2 s.t. G_1 is add & G_2 is even & we obtain the following approximation.[2]

$$J(y) \cong K_1\left(E\{G_1(y)\}\right)^2 + K_2\{E\{G_2(y)\} - E\{G_2(U)\}\}^2 \quad \ldots(9)$$

Where K_1 & K_2 are positive constant & U is standardized gaussian variable.

4 Fast Fixed Point Algorithm Using Negentropy

As with kurtosis, a much faster method for maximizing negentropy than that given by the gradient method can be found using a fixed-point algorithm. This algorithm finds a direction i.e., unit vector **W** such that the projection $\mathbf{W}^T\mathbf{Z}$ maximizes non-gaussianity. Non gaussianity is measured by the approximation of negentropy J ($\mathbf{W}^T\mathbf{Z}$) , where the variance of $\mathbf{W}^T\mathbf{Z}$ must be constrained to unity , for whitened data this is equivalent to constraining the normalization of **W** to be unity.[5,6]

Fast ICA is based on a fixed-point iteration scheme for finding a maximum of the non-guassianity of $\mathbf{W}^T\mathbf{Z}$. It can be derived as an approximative Newton iteration. The fast ICA algorithm using negentropy combines the superior algorithmic properties resulting from the fixed-point iteration with the preferable statistical properties due to negentropy. Considering the algorithm stated in [1], we modified the Fast ICA algorithm using Negentropy is as follows:

1. Center the data to make its mean zero.

2. Choose m, the number of independent components to estimate from the PCA.

3.. Whiten the data to give **Z**.

4. Choose the random mixing matrix **W**

5. Orthogonalized the matrix W

6. Let $\mathbf{W_1} \leftarrow \mathbf{E\{Zg(W^TZ)\} - E\{g'(W^TZ)\}W}$,
   where g is defined as
   g(y)= tanh(y) or
   $g(y)=y^3$

7 Orthogonalized matrix **W**

8. If not converged, go back to step 6.

9. Let $\mathbf{W}_2 \leftarrow \mathbf{W}_1/||\mathbf{W}_1||$

10. for second ICA go to step 6

11. Repeat for i= 1,2,3….m

Here convergence means that the old and new values of **W** point in the same direction i.e., the absolute value of their dot product is (almost) equal to one.

5. Result and Conclusion

We have taken different images for training. The code was written and simulated in MATLAB. Mixing matrix was assumed to be random. Hence every time we run the algorithm the mixed image coefficient was different. We consider the image of size 50 x 50. The code has been simulated many times out of which we are giving data for first 10 iterations.

Table 1 shows the total number of iterations taken by each non-linearity to converge and the error estimated between initially assumed **S** and the **S** what we got from ICA. Figure 1 shows the estimated error analysis between the initial mixing matrix assumed and Ica found out. Figure 2 shows the number of iterations taken by each nonlinearity. Figure 3. Shows the plot of ICA we got for images using tanh. Figure 4. Shows the plot of ICA we got for images using y^3. Figure 5 to Figure 11 shows the intermediate results of the simulation. Figure 12 shows the images given as the input to the mixing matrix to get the mixed image i.e Fig 13. Figure 14 shows the images retrieved using nonlinearity tanh. Figure 15 shows the images retrieved using nonlinearity y^3.

The algorithm was tested on several set of mixed images and the following conclusions were drawn The number of images mixed can be understood by seeing at PCA output. The number of PCA will give the number of images that are mixed in the given input mixed matrix.

Depending of dewhitening matrix, the image is obtained. If the maximum dewhiteneing coefficient lie in second and fourth quadrant, then original image is retrieve otherwise image retrieve is negative.

Figure 7 and Figure 10 proves the convergence of algorithm because it is as per the definition of convergence of kurtosis, that the convergence takes place in the negative direction, where the maxima are at the points when exactly one of the element vectors of q is zero and other non-zero. Also it can be seen from the convergence graph that non zero element lie between + constant value to – constant value.

Fig 16 to Fig 23 shows the result for next set of mixed images, It shows that if set of mixed images are such that, there are two or more images are having gaussian histogram distribution then, even though the convergence is there but retrieval of images is not good. Thus it support to the basic Restriction of ICA that it is essentially difficult if the observed variables have gaussian variable and hence not more that one gaussian variable is allowed.

The retrieve images are not in the same order every time we run the algorithm and hence it is difficult to determine the order of the independent components

It is found that the number of iterations required to converge the algorithm using nonlinearity Y^3 is less as compared to the tanh nonlinearity. But it is at the expense of probability of getting Independent Component. The estimated error graph shows that the error is more while retrieving images with Y^3 as compared to tanh nonlinearity. Also some post processing steps may require for Y^3 because sometimes negative images get retrieved. Hence we conclude that for finding the ICA the nonlinearity which varies slowly will give better results.

Table.1

Non linearity	run	no. of iteration	Means square error			
			I1	I2	I3	I4
tanh	1	32	0.0020	0.0029	0.0067	0.0023
tanh	2	14	0.0015	0.0023	0.0081	0.0020
tanh	3	8	0.0024	0.0015	0.0080	0.0020
tanh	4	34	0.0064	0.0030	0.0019	0.0024
tanh	5	9	0.0015	0.0080	0.0021	0.0024
tanh	6	40	0.0068	0.0022	0.0029	0.0019
tanh	7	44	0.0019	0.0023	0.0065	0.0030
tanh	8	47	0.0029	0.0023	0.0020	0.0066
tanh	9	55	0.0019	0.0066	0.0023	0.0031
tanh	10	15	0.0024	0.0015	0.0079	0.0021
pow3	1	57	0.0068	0.0051	0.0011	0.0009
pow4	2	17	0.0049	0.0062	0.0010	0.0020
pow5	3	18	0.0019	0.0010	0.0061	0.0047
pow6	4	19	0.0010	0.0020	0.0060	0.0048
pow7	5	13	0.0047	0.0061	0.0020	0.0010
pow8	6	19	0.0010	0.0020	0.0062	0.0048
pow9	7	14	0.0011	0.0047	0.0062	0.0021
pow10	8	20	0.0046	0.0010	0.0020	0.0063
pow11	9	21	0.0010	0.0047	0.0063	0.0020
pow12	10	18	0.0047	0.0060	0.0010	0.0021

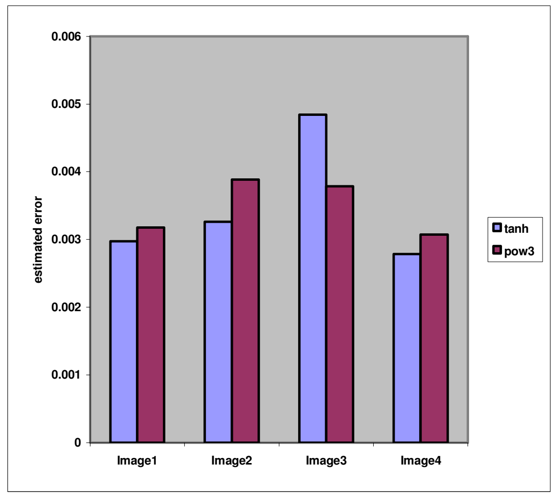

Fig.1 Analysis of estimated error

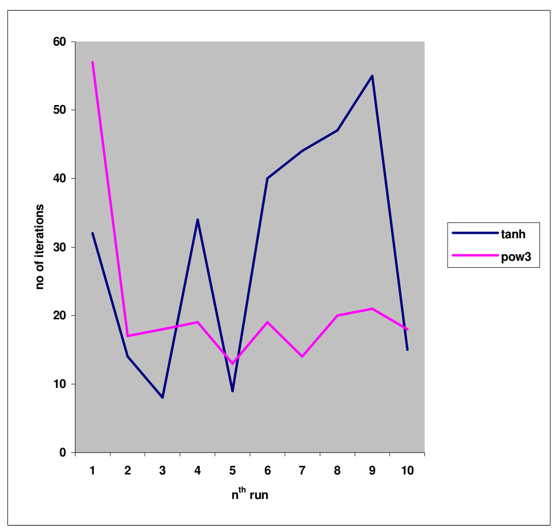

Fig.2. No. of iteration required for convergence

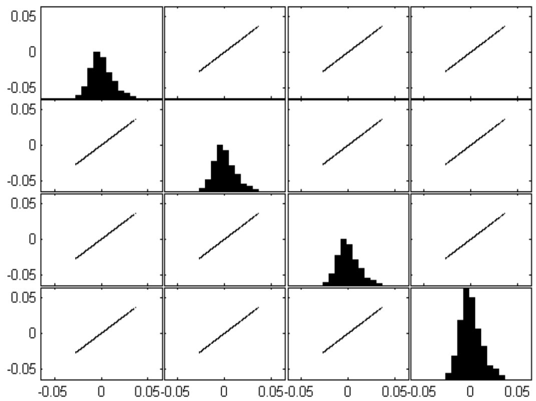

Fig 3.ICA and histogram plot for images using tanh

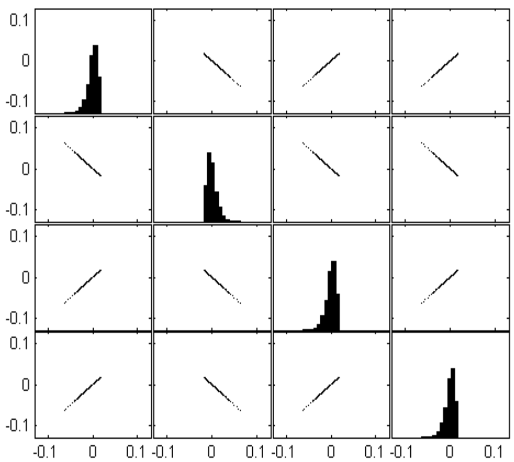

Fig 4. ICA and histogram plot for image 4 using y3

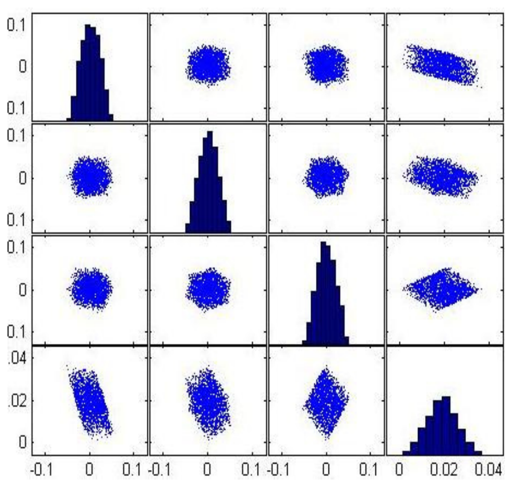

Fig 5 PCA output

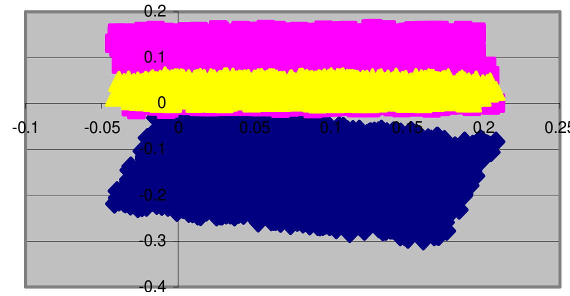

Fig 6 Whitening for tanh nonlinearity

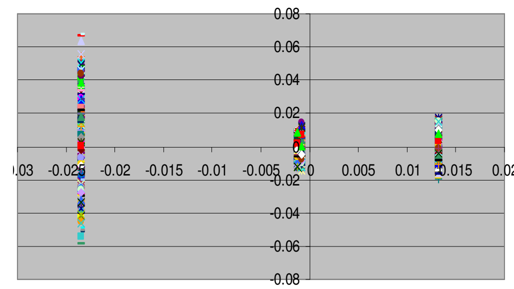

Fig 7 Plot of convergence for tanh nonlinearity

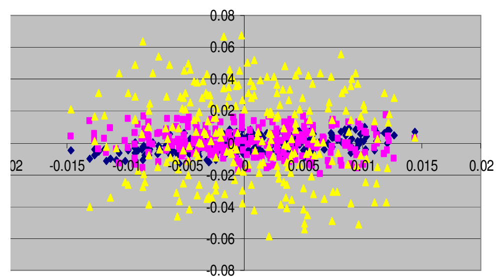

Fig 8 Dewhitened matrix for tanh nonlinearity

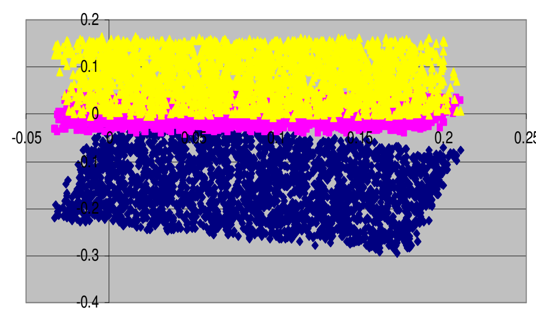

Fig 9 Whitening for y^3 nonlinearity

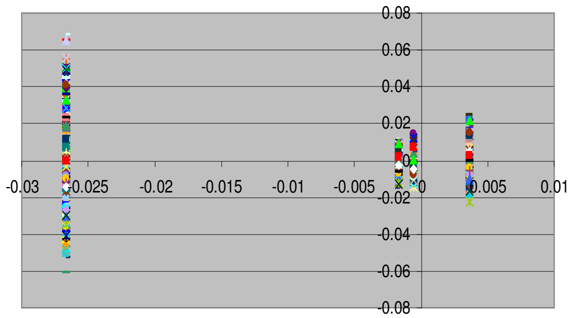

Fig 10 Plot of convergence for y^3 nonlinearity

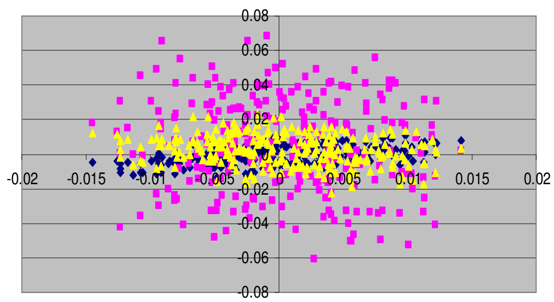

Fig 11 Dewhitened matrix for y^3 nonlinearity

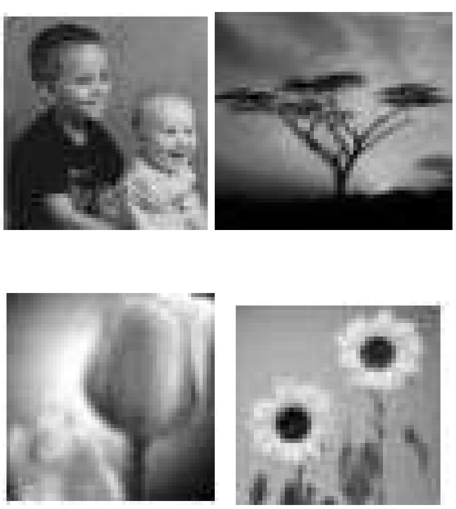

Fig.12.Original Input Images

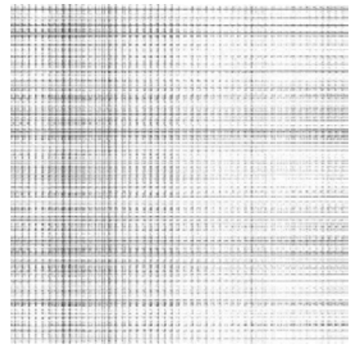

Fig 13. Mixed image given as an input to the algorithm

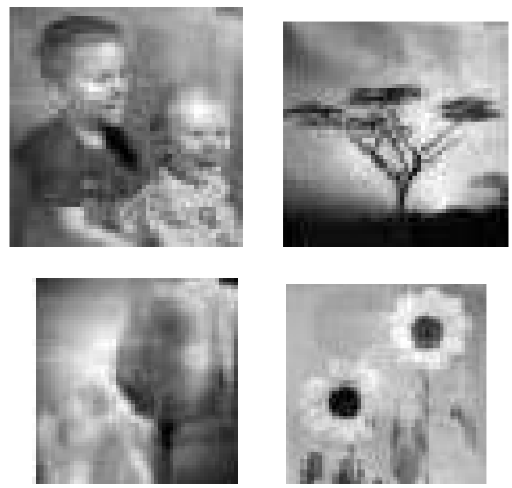

Fig.14.Output Images for Tanh nonlinearity

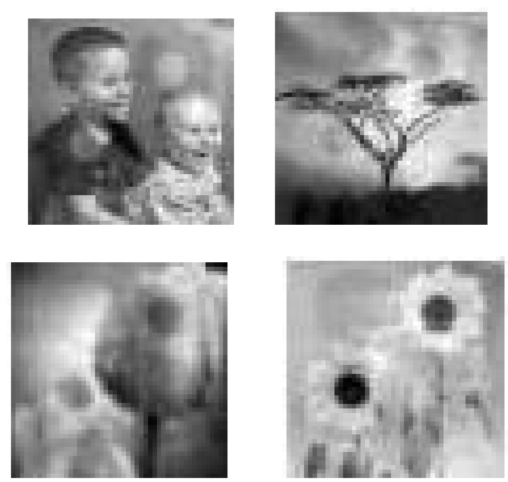

Fig.15.Output Images For Y^3 nonlinearity

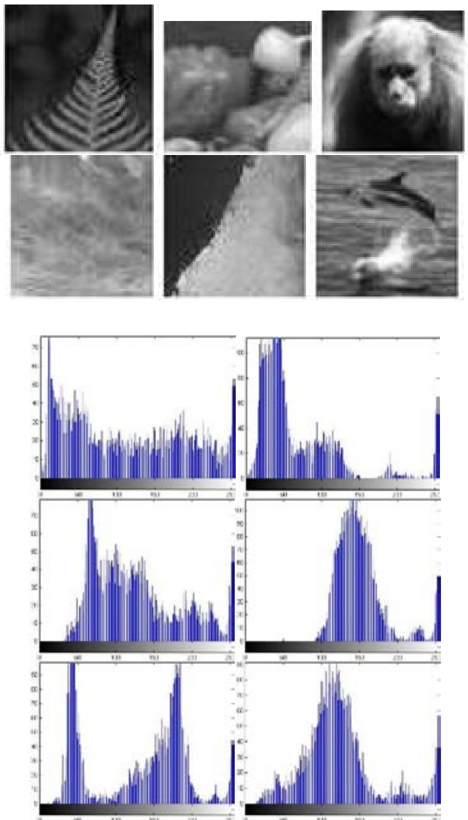

Fig. 16 Input images with their histogram

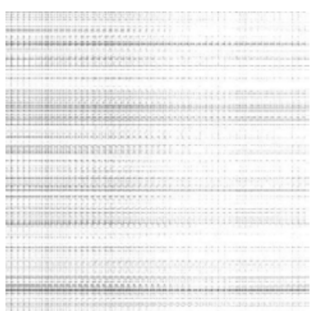

Fig 17. Mixed image given as an input to the algorithm

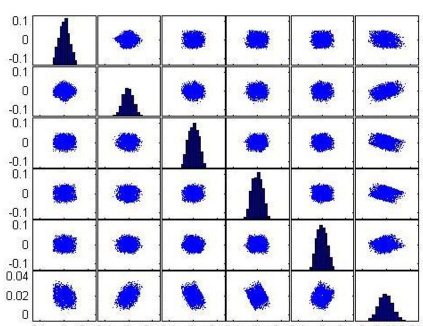

Fig 18. PCA output

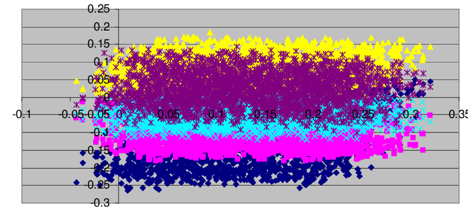

Fig 18 Whitening for tanh nonlinearity

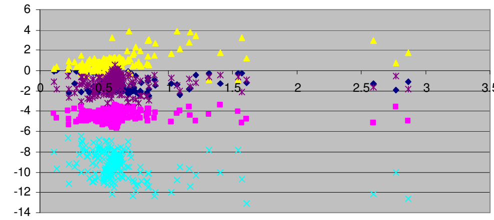

Fig 19. One dimensional plot for ICA

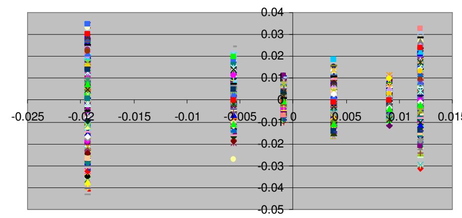

Fig 20. Plot of convergence for tanh nonlinearity

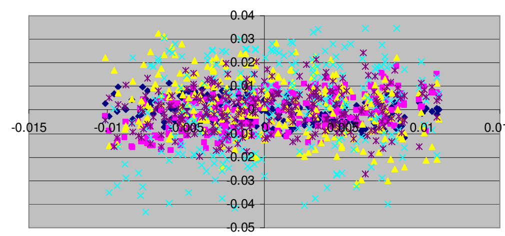

Fig 21 Dewhitened matrix for tanh nonlinearity

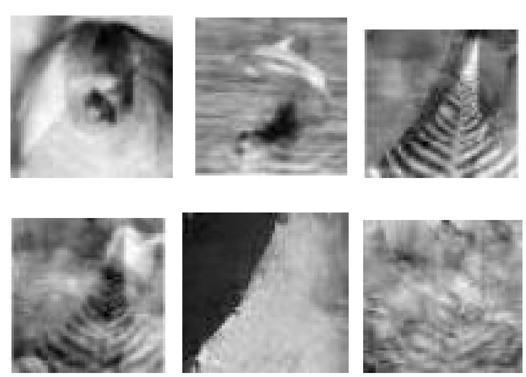

Fig 22 Output for tanh non-linearity

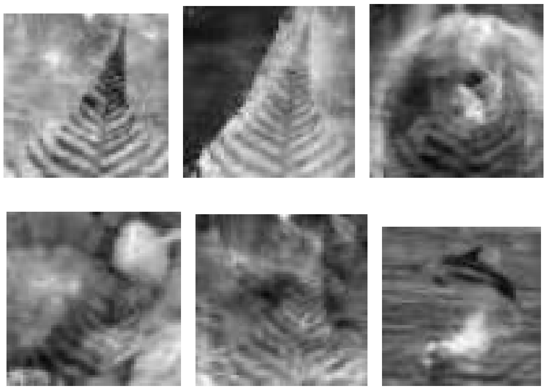

Fig 23 Output for y^3 non-linearity

Acknowledgement: We are very much thankful to Prof. B.L.Deekshtulu, ISRO Visiting Professor, HCU and Ex-director & distinguished Scientist, NRSA, Prof Nahari Pandit Professor of Statistics, ATRI and Prof. Bapiraju, Dept. of AI, HCU for helping us in understanding the concept of ICA.

References:

[1] A. Hyvarinen. Survey of Independent Component Analysis. *Neural Computing Surveys*, 2:94-128, 1999.

[2] A.Hyvarinen and E.Oja. Independent component analysis: Algorithms and application. *Neural Networks*, 13(4-5):411- 430, 2000.

[3] A. Hyvarinen. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Trans. on Neural Networks,* 10(3): 626- 634, 1999

[4] A. Hyvarinen. A family of fixed-point algorithms for independent component analysis. *Proc.IEEE Int. conf. on Acoustics, Speech and Signal Processing* (ICASSP'97), pages 3917- 3920, Munich, Germany, 1997.

[5] A. Hyvarinen and E. Oja. A fast fixed-point algorithm for independent component analysis. *Neural Computation*, 9(7): 1483- 1492, 1997.

[6] A. Hyvarinen, R.Cristescu, and E.Oja. A fast algorithm for estimating overcomplete ICA bases for image windows. *Proc.Int.Joint Conf. on Neural Networks*, Pages 894-899, Washington, D.C., 1999.

[7] T.W.Lee. Independent Component analysis – Theory and Applications. *Kluwer*, 1998.

[8] A.J.Bell and T.J.Sejnowski. The 'independent components' of natural scenes are edge filters. *Vision Research*, 37:3327-3338, 1997.

[9] A.J. Bell. Information theory, independent component analysis, and applications. *S.Haykin editor, Unsupervised Adaptive Filtering*, Vol.I, pages 237-264. Wiley, 2000.

[10] P. Comon. Independent component analysis—a new concept? *Signal Processing*, 36:287-314, 1994.

[11] P.Comon and P.Chevalier. Blind source separation: Models, concepts, algorithms and performance. *In S.Haykin, editor, Unsupervised Adaptive Filtering*, Vol.I pages 191-235. Wiley, 2000.

[12] R.M.Everson and S.Roberts. Independent component analysis: a flexible nonlinearity and decorrelating manifold approach. N*eural Computation*, 11(8): 1957-1983, 1999

[13] A.Hayvarinen and E.Oja. Independent Component analysis by general nonlinear Hebbian-like learning rules. *Signal Processing*, 64(3): 301-313, 1998.

[14] Feng Shao, Gangyi Jiang* and Mei Yu Color Correction For Multi-View Images Combined With PCA And ICA *WSEAS Transactions on Biology and Biomedicine* Issue 6, Volume 4, June 2007

[15] P. Gao, W. L. Woo, S. S. Dlay Nonlinear Independent Component Analysis Using Generalised Polynomial Neural Network**,** *WSEAS Transactions On Information Science And Applications,* Issue 6, Volume 1, December 2004

[16] P. Gao, W. L. Woo, S. S. Dlay, A FOCUSS-Based Algorithm for Nonlinear Overcomplete Independent Component Analysis, *WSEAS Transactions On Information Science And Applications*,Issue 6, Volume 1, Dec 2004