Modeling Neural Networks and Curvelet Thresholding for Denoising Gaussian noise

B. Bhosale

S H Kelkar College of Arts, Commerce and Science, University of Mumbai, Devgad 416613 (M.S.), India
Email: bnbhosale15@gmail.com

Abstract: Modeling facilitates simulating the behavior of a system for a variety of initial conditions, excitations and systems configurations. The quality and the degree of the approximation of the models are determined and validated against experimental measurements. Neural networks are very sophisticated modeling techniques capable of modeling extremely complex functions employed in statistics, cognitive psychology and artificial intelligence. Especially, neural network model that emulates the central nervous system is a part of theoretical as well as computational neuroscience. Since, graphs are the abstract representation of the neural networks, graph analysis has been used in the study of neural networks, anatomical and functional connectivity. This approach has given rise to a new representation of neural networks, called Graph neural network, which encompasses both Biological neural networks and Artificial neural networks. In machine learning and cognitive science, a class of artificial neural networks is a family of statistical learning algorithms inspired by biological neural networks (the central nervous systems of animals, in particular, the brain) and they are used to estimate or approximate functions that depend on a large number of unknown inputs. Particularly, in the area of signal processing, the prime goal of the neural networks is to obtain a good approximation for some input/output mapping, by enhancing the performance of captured signals. Consequently, improving quality of noisy signals/images through noise reduction has become an active area of research. In such noise reduction problems, the graph neural networks can be used quite effectively by suitably designing it to train with input sequences; which are assumed to be a composition of the desired signal plus an additive Random/Gaussian noise.

Until recently, wavelets had been widely used in signal processing. However, wavelets suffer from the limitations of orientation selectivity and as a result, they fail to represent changing geometric features of the signal along edges effectively. Curvelet transform, on the contrary, exhibits good reconstruction of the edge data as it incorporates a directional component to the conventional wavelet transform and therefore can be robustly used in analysis of higher dimensional signals.

In this paper, the endeavour is to employ graph neural networks in combination with the curvelet transform for the enhancement of the corrupted signal and device a unified signal denoising technique using appropriate thresholding function. The experimental results show that the proposed model produces better results in terms of signal to noise ratio and computed mean square error - the most commonly used measures to determine performance factor/quality of the captured signal.

Keywords: *Graph neural networks, curvelet transform, thresholding function, signal denoising*

1.   INTRODUCTION

The advancement in the theory of neural networks has inspired new applications in the field of neuroscience. These applications include study of anatomical and functional connectivity based upon functional magnetic resonance imaging, electroencephalography and magnetoencephalography (Cornelis et al., 2007). Of late, wavelet based multi-resolution techniques were widely used in image/signal processing, optical data analysis etc. Among others, applications of wavelet transform has been explored in analyzing solitons (Bhosale and Biswas, 2013), bio-informatics (Bhosale et al., 2013), neural networks (Bhosale et al., 2013). However, wavelets fail to provide good direction selectivity, which is also an important response property of simple cells and neurons at stages of the visual pathway. To overcome this limitation, in 1999, an anisotropic geometric wavelet transform, named ridgelet transform, was proposed by Candes and Donoho, which later on modified to block ridgelet-based transform called curvelet transform (Candès and Donoho, 2005). Subsequently, a considerably simpler second-generation curvelet transform based on a frequency partition technique was proposed by the same authors. This modified second-generation curvelet transform has been proved to be a versatile and efficient tool for signal/image processing, seismic data exploration, fluid mechanics, and solving partial differential equations encountered in non-linear physical phenomena. Recently, curvelet transform has been used in image denoising for its ability to obtain high quality images (Kota and Reddi, 2011). From the analysis of CT scan images, curvelet transform shows better results in terms of recovering the original image from the noisy one than the wavelet transform (Sivkumar, 2007). As most of the natural signals are assumed to have additive random noise, modeled as Gaussians, removing additive Gaussian noise by nonlinear methods such as curvelet denoising has produced better results (Yaser and Mahdi, 2011). Besides, curvelets in combination with neural networks have been successfully used in seismic data processing (Hassan et al., 2013). Another study presented an integrated classification machine (ICM), which is a hierarchy of artificial neural networks that are trained to classify the seismic waveforms (Shimshoni and Intrator, 1998). A velocity model inversion approach using artificial neural networks for the study of aftershocks from the 2000 Tottori, Japan, earthquake located around station SMNH01was proposed by Moya, (2010).

Various techniques are reportedly used in signal extraction and related applications. Prominent amongst them are: one, separating noisy mixed signals using fast Independent Component Analysis (ICA) algorithm and then applying curvelet thresholding and the other one, using neural network thresholding to denoise the mixed signals. The present work elucidates a systematic transition from classical wavelets to curvelets; and proposes a model that unifies both these approaches.

2.   CURVELET TRANSFORM FORMULATION

This formulation is due to Candes and Donoho (2005), which involves two main components: one, considering polar coordinates in frequency domain and two, constructing curvelet elements being locally supported near wedges.

The Continuous Curvelet Transform (CCT), $f \rightarrow \Gamma_f(a, b, \theta)$, of functions $f(x_1, x_2)$ on R^2, into a transform domain with continuous scale $a > 0$, location $b \in R^2$, and orientation $\theta \in [0, 2\pi)$ is devised as follows:

Consider the polar coordinates in frequency domain, $r = \sqrt{\xi_1^2 + \xi_2^2}$, $\omega = arctan\frac{\xi_1}{\xi_2}$ corresponding to $x = (x_1; x_2)^T$, the spatial variable, and $\xi = (\xi_1, \xi_2)^T$, the variable in frequency domain. Also consider the pair of windows $W(r)$ and $V(t)$, called radial window and angular window respectively

$$V(t) = \begin{cases} 1, |t| \leq 1/3 \\ \cos[\pi/2\nu(3|t| - 1)], 1/3 \leq |t| \leq 2/3 \\ 0, else \end{cases}$$

$$W(r) = \begin{cases} \cos[\pi/2\nu(5 - 6r)], 2/3 \leq r \leq 5/6 \\ 1, 5/6 \leq r \leq 4/3 \\ \cos[\pi/2\nu(3r - 4)], 4/3 \leq r \leq 5/3, 0, else \end{cases} \tag{1}$$

where ν is a smooth function satisfying

$$\nu(x) = \begin{cases} 0, x \leq 0 \\ 1, x \geq 1 \end{cases}, \nu(x) + \nu(1 - x) = 1, x \in R$$

The windows should satisfy the admissibility conditions (2)

$$\int_0^\infty W(ar)^2 \frac{da}{a} = 1, \forall r > 0, \int_{-1}^1 V(t)^2 dt = 1$$
(2)

The window functions are used to construct a family of complex-valued waveforms, called curvelets, with three parameters: the *scale* $a \in (0; 1]$, the *location* $b \in R^2$, and the *orientation* $\theta \in [0; 2\pi)$. Using polar coordinates $(r; \omega)$ in frequency domain, a-scaled window, called polar wedge, is given as

$U_a = a^{3/4} W(ar) V\left(\frac{\omega}{\sqrt{a}}\right)$, for some a with $0 < a \le 1$, where the support of U_a depends on the supports of windows W and V.

Let the basic element $\gamma_{a,0,0} \in L^2(R^2)$ be given by its Fourier transform as $\hat{\gamma}_{a,0,0} = U_a(\xi)$.

By translation and rotation of basic element $\gamma_{a,0,0}$, the family of analyzing elements, called curvelets, is generated as

$$\gamma_{ab\theta}(x) = \gamma_{a00}(R_\theta(x - b))$$
(3)

where R_θ is the 2×2 rotation matrix effecting planar rotation by θ radians

$$R_\theta = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$
(4)

Applying this family of high frequency elements, $\{\gamma_{a,b,\theta}: (0,1], b \in R^2, \theta \in [0,2\pi\}$, CCT, Γ_f of $f \in L^2(R^2)$, the curvelet transform is expressed as

$$\Gamma_f(a, b, \theta) = \langle \gamma_{a,b,\theta}, f \rangle = \int_{R^2} \gamma_{a,b,\theta}(x)\overline{f(x)}dx$$
(5)

The corresponding inversion/reproducible formula is

$$f(x) = \int \Gamma_f(a, b, \theta)\, \gamma_{ab\theta}(x)\mu(da\, db\, d\theta)$$
(6)
$$\|f\|_{L^2}^2 = \int |\Gamma_f(a, b, \theta)|^2 \mu(da\, db\, d\theta)$$

where, μ denotes the reference measure, $d\mu = \frac{da}{a^3} db\, d\theta$

This formula is valid for $f \in L^2$ that has a Fourier transform vanishing for $|\xi| < 2/a_0$.

A discrete version of the continuous curvelet transform, called Discrete curvelet transform (DCT), can be derived by a suitable sampling at the range of scales, orientations and locations (Kota and Reddi, 2011).

To construct DCT, choose the scales $a_j = 2^{-j}, j \ge 0$; the equidistant sequence of rotation angles

$\theta_{j,l}, \theta_{j,l} = \frac{\pi l 2^{-[j/2]}}{2}$ with $l = 0,1,...,4.\, 2^{[j/2]} - 1$; and the positions

$b_k^{j,l} = b_{k_1,k_2}^{j,l} = R_{\theta_{j,l}}^{-1}\left(\frac{k_1}{2^j}, \frac{k_2}{2^{j/2}}\right)$, with $k_1, k_2 \in Z$.

This choice will lead to a discrete curvelet system that forms a tight frame.

In this system, every function $f \in L^2(R^2)$ will be representable by a curvelet series,

$f = \sum_{j,k,l} c_{j,k,l}(f)\gamma_{j,k,l} = \sum_{j,k,l}\langle f, \gamma_{j,k,l}\rangle\gamma_{j,k,l}$, for which the Parseval identity,

$\sum_{j,k,l}|\langle f, \gamma_{j,k,l}\rangle|^2 = \|f\|_{L^2(R^2)}^2, \forall f \in L^2(R^2)$ holds.

The terms in the above series, $c_{j,k,l}(f) = \langle f, \gamma_{j,k,l}\rangle$ are the curvelet coefficients.

As a matter of fact, the curvelet coefficients are obtained by Plancherel's theorem for $j \ge 0$ as

$$c_{j,k,l}(f) = \int_{R^2} f(x)\overline{\gamma_{j,k,l}}(x)dx = \int_{R^2} \hat{f}(\xi)\overline{\hat{\gamma}_{j,k,l}}(\xi)d\xi = \int_{R^2} \hat{f}(\xi)U_j\left(R_{\theta_{j,l}}\xi\right)e^{i\langle b_k^{j,l},\xi\rangle}d\xi,$$
(7)

where $\hat{f}(\xi) = \frac{1}{2\pi}\int_{R^2} f(x)e^{-i\langle x,\xi\rangle}dx$.

The family of curvelet functions in this system, $\gamma_{j,k,l}(x)$, is generated from the basic curvelet

$\hat{\gamma}_{j,0,0}(\xi) = U_j(\xi)$ as

$\gamma_{j,k,l}(x) = \gamma_{j,0,0}\left(R_{\theta_{j,l}}\left(x - b_k^{j,l}\right)\right)$, where the scaled windows take the form

$$U_j(r,\omega) = 2^{-3j/4} W(2^{-j}r)V\left(\frac{2^{j/2}\omega}{2\pi}\right) = 2^{-3j/4} W(2^{-j}r)V\left(\frac{\omega}{\theta_{j,l}}\right) \tag{8}$$

It is seen that the curvelet functions, $\gamma_{j,k,l}(x)$, in frequency domain are obtained as the inverse Fourier transform of a suitable product of the windows W and V as

$$\hat{\gamma}_{j,k,l}(\xi) = e^{-i\langle b_k^{j,l},\xi\rangle}U_j\left(R_{\theta_{j,l}},\xi\right) = e^{-i\langle b_k^{j,l},\xi\rangle}2^{-3j/4} W(2^{-j}r)V\left(\frac{\omega+\theta_{j,l}}{\theta_{j,l}}\right) \tag{9}$$

In practical implementations, Cartesian arrays instead of the polar tiling of the frequency plane are convenient to use. The Cartesian counterpart of curvelet-like functions is given by

$\gamma_{j,k,l}(x) = \gamma_{j,0,0}\left(S_{\theta_{j,l}}\left(x - b_k^{j,l}\right)\right)$, which is generated from the basic curvlets, $\hat{\gamma}_{j,0,0}(x) = U_j(\xi)$

with shear matrix $S_\theta = \begin{pmatrix} 1 & 0 \\ -\tan\theta & 1 \end{pmatrix}$.

The curvelet coefficient in Cartesian arrays will be therefore obtained as

$$c_{j,k,l}(f) = \int_{R^2} \hat{f}(\xi)U_j\left(S^{-1}_{\theta_{j,l}}\xi\right)e^{i\langle b_k^{j,l},\xi\rangle}d\xi = \int_{R^2} \hat{f}\left(S_{\theta_{j,l}}\xi\right)U_j(\xi)e^{i\langle k_j,\xi\rangle}d\xi, \tag{10}$$

where $k_j = \left(k_1 2^{-j}, k_2 2^{-[-j/2]}\right), (k_1, k_2)^T \in \mathbb{Z}^2$

Yet another version, called Fast digital curvelet transform (FDCT), is based on wrapping of Fourier samples (Candes et al. 2006). This takes a 2D signal/image as an input in the form of a Cartesian array $f[m,n]$, where $0 \le m < M, 0 \le n < N$ where M and N are the dimensions of the array; and the output will be a collection of curvelet coefficients, $c^D(j, l, k_1 k_2)$, indexed by a scale j, an orientation l and spatial location parameters k_1 and k_2, given as

$$c^D(j, l, k_1 k_2) = \sum_{\substack{0 \le m \le M \\ 0 \le n \le N}} f[m,n]\gamma_{j,k,k_1 k_2}^D[m,n], \tag{11}$$

for any bi-variate function, $f \in L^2(R^2)$ associated with signal vector.

3. NEURAL NETWORKS

A pulse, or a signal or an image can be represented by neural network in abstract form. Neural networks are models which are structurally very simple because they consist of very simple building blocks, neurons that are organized into layers. A neuron is a basic element of a network that is mathematically presented as a point in space toward which signals are transmitted from surrounding neurons or inputs. Since graphs are the abstract representation of neural networks, a Graph theoretic approach can be invoked wherein the computational elements, neurons of the network can be shown as nodes. Recently various approaches have been unified in neural network models under graph neural networks (GNN), which are used for processing the data represented in graph domains. GNN is a larger class of neural networks which includes Biological neural networks (BNN) and Artificial neural networks (ANN) (Podolak, 1998), which are widely used in noise reduction problems (Badri, 2010). Moreover, GNNs have been proved to be a sort of universal approximator for functions on graphs and have been applied to several problems, including spam detection, object localization in images, molecule classification (Scarcelli, 2009). In mathematical sense, GNN implements a function $\tau(G, n) \in R^m$ that maps a graph G and one of its nodes n onto an m-dimensional Euclidean space.

4. LINEAR GAUSSIAN FACTOR ANALYSIS

Independent component analysis (ICA) is widely used in statistical signal processing, medical image processing, and telecommunication applications (Hyvrinen and Oja, 2000). For example, ICA is used for finding factors or components from multivariate statistical data and is one of the many solutions to the blind source separation (BSS) problem (Cardoso and Souloumiac, 1993). BSS is a method of extracting underlying source signals from a set of observed signal mixtures with little or no information to the nature of these source signals (Shehata et al., 2013).
The basic ICA model is expressed as,

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{\eta}(t) \tag{12}$$

where $\mathbf{x}(t)$ is an *N* dimensional vector of the observed signals at the discrete time instant *t*, A is an unknown mixing matrix, $\mathbf{s}(t)$ is original source signal of $M \times N$ $(M \leq N)$ and $\mathbf{\eta}(t)$ is the observed noise vector and *M* is number of sources.

More explicit ICA model, called linear Gaussian factor analysis (GFA) model that represents the signal by GNN (Valpola, 2000) is,

$$x_i(t) = \sum_j A_{ij} s_j(t) + a_i + \eta_i(t) \tag{13}$$

where i indexes different components of the observation vector x_i, representing weighted sums of underlying latent variables, j indexes different factors and A_{ij} are the weights of the factors s, also known as factor loadings. The factors *s* and noise η are assumed to have zero mean. The bias in x is assumed to be caused by a. The effect of the inaccuracies and other causes is summarized by Gaussian noise η. If the variances of the Gaussian noise terms, $\eta(t)$ are denoted by σ_i^2, the probability which the model gives for the observation $x_i(t)$ can be written as

$$P(x_i(t) \mid s(t), \mathbf{A}, a_i, \sigma_i^2) = \frac{1}{\sqrt{2\pi\sigma_i^2}} exp\left(-\frac{[x_i(t) - \sum_j A_{ij} s_j(t) - a_i]^2}{2\sigma_i^2}\right) \tag{14}$$

In vector form, GFA can be expressed as

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{a} + \mathbf{\eta}(t) \tag{15}$$

Equivalently, $\mathbf{x}(t) \sim N(\mathbf{As} + \mathbf{a}, \mathbf{\sigma}^2)$, where x, s, a and η are vectors and A is a matrix.

The purpose of the component analysis is to estimate original signal, $\mathbf{s}(t)$, from the mixed signal, $\mathbf{x}(t)$. That is, equivalent to estimating the matrix A. Assuming that there is a matrix W, which is the de-mixing matrix or separation inverse matrix of A, then the original source signal is obtained by

$$\mathbf{s}(t) = \mathbf{W}\mathbf{x}(t) \tag{16}$$

This ICA algorithm assumes that the mixing matrix A must be of full column rank and all the independent components $\mathbf{s(t)}$, with the possible exception of one component, must be non-Gaussian (Sivkumar, 2007).

5. CURVELET INDUCED SIGNAL DENOISING

In physical systems, transmitted signals are usually distributed partially, or sometimes almost completely by an additive noise from the transmitter, channel, and receiver. Noise arises due to imperfect instruments used in signal processing, problems with the data acquisition process, and interference which can degrade the data of interest. Also, noise can be introduced due to compression and transmission errors (Motwani et al., 2004). De-noising or the noise reduction is an essentially required process to enhance the estimation process of signal/image reconstruction of the captured signal. Several signal denoising techniques are proposed in reported literture.

Among others, curvelets, being analytic signal filters, curvelet thresholding technique has been prominently used in signal processing. Curvelet thresholding is a simple operation, which aims at reducing noise in a noisy signal. It is performed by selecting the FDCT coefficients below a certain threshold and setting them to zero

$$y_\lambda = \begin{cases} y_\lambda, & |y_\lambda| \geq t_\lambda \\ 0, & |y_\lambda| < t_\lambda \end{cases} \tag{17}$$

where t_λ is the threshold and λ is the index.

The threshold used in this technique is $t_\lambda = k\sigma_\lambda\sigma,$ for some scale k, where σ is an estimation of the standard deviation of the noise estimated from the finest scale coefficients.

Another approach adopted recently in signal denoising problems is thresholding neural network (TNN), wherein thresholding function is used instead of activation functions in feed forward neural network and threshold value is adjusted using (Yaser and Mahdi, 2011),

$$t_{\lambda+1} = t_\lambda + \Delta t_\lambda \text{ , where } \Delta t_\lambda = -\propto \left.\frac{\partial MSE}{\partial t}\right|_{t=t_\lambda}, \propto \text{ is learning rate,} \tag{18}$$

t_λ is a threshold, $t_\lambda = \sigma\sqrt{2log\ L}$, σ is a noise variance and $L = N^2$ is size of the signal.

Now, in the proposed technique, recalling linear Gaussian factor analysis (GFA) model that represents the signal $x_i(t)$ by GNN (13) and applying FDCT (11) to signal, $x_i(t)$, to obtain curvelet coefficients:

$$x^D(j, l, k_1 k_2) = \sum_{0 \leq n \leq N}^{0 \leq m \leq M} x_i[m, n]\gamma_{j,k,k_1 k_2}^D[m, n] \tag{19}$$

The expression (19) represents the decomposition of the signal into a sequence of curvelet coefficients.

The curvelet coefficients of noisy signal so obtained can be now used as the input to the threshold function,

$$f(x, t_\lambda) = x - \frac{x}{exp[(x/t_\lambda)^2 - 1]} + \frac{1/8x}{exp(x/0.71t_\lambda)^2} \tag{20}$$

where x depicts the curvelet coefficients obtained from noisy signal $x_i(t)$.

The thresholding function $f(x, t_\lambda) \to 0$ in $[-t_\lambda, t_\lambda]$, and $f(x, t_\lambda) = x$ outside the threshold interval, thus, converging to curvelet coefficients themselves. It means, the noisy coefficients in the threshold interval $[-t_\lambda, t_\lambda]$ are shrieked. For chosen learning rate and convergence value, the universal threshold value (18) is obtained during learning process. In test phase, computed threshold value in learning phase is used by the thresholding function (20) to demise curvelet coefficients of test signals.

Finally, inverse curvelet transform (6) is applied to get the desired denoised signal $\hat{x}_i$.

The proposed curvelet induced GFA signal denoising model can be put to performance factor analysis. Among the various performance factors, the peak signal to noise ratio (PSNR) and root mean square error (RMSE) are the most commonly used measures of quality of reconstruction in signal denoising. Higher the value of PSNR with minimum value of RMSE, better the performance of the denoising model.

The PSNR and RMSE are calculated using the expressions

$$PSNR = 10\log_{10}\left(\frac{\sum_{i=1}^{N} x^2(i)}{\sum_{i=1}^{N}[x(i) - \hat{x}(i)]^2}\right), RMSE = \frac{1}{N}\sqrt{\sum_{i=1}^{N}[x(i) - \hat{x}(i)]^2} \tag{21}$$

where $x(i)$ is the original source signal, $\hat{x}(i)$ is the separated signal, i is the sample index and N is the number of samples of the signal.

The PSNR and the other statistical measures such as mean and standard deviation of the PSNR for different signal sub-bands/samples can be compared to determine the performance of the proposed signal denoising model.

To illustrate the performance of the devised GFA model, simulations are performed on noisy mixed 256X256 'Lena' image and sample CT image on Matlab® R 7.9 on a core i7 2.2 GHz PC and curvelet transform via USFFT software package.

The denoising results (PSNR in dB) in both the experiments are presented below:

Image	Noisy Image	Soft curvelet thresholding	TNN thresholding	Curvelet induced GFA	RMSE
Lena	20.01(σ=10)	24.89	32.05	34.25	0.177533
CT (Random noise)	21.09 (m=0, σ^2=0.01)	26.89	28.74	32.14	0.164434
CT (Gaussian noise)	19.77(m=0, σ^2=0.01)	27.70	29.64	31.17	0.14564

From the experimental results, it is seen that the curvelet induced GFA gives increased PNSR with minimum RMSE for the used images. Similar simulations can be performed for satellite images, seismic data, medical images.

6.   CONCLUSION

The present work integrates the Independent component analysis, graph neural networks and curvelet thresholding techniques adopted in signal denoising processes so as to harvest the advantages and overcome the inherent limitations of each of them. Denoising using the curvelet transform recovers the original signal from the noisy one using lesser coefficients than the wavelet transform. Moreover, curvelet transform provides high PSNR- the most commonly used measure of quality of reconstructed signal- and removes the Random as well as Gaussian white noises. Further, the wrapping based Curvelet transform is conceptually simpler, faster and invertible with rapid inversion algorithm of the same complexity than the existing techniques. Thus curvelet outperforms the wavelet transform in several ways.

The proposed curvelet induced Gaussian factor analysis model in Graph neural network settings unifies earlier denoising models, consequently enhances the performance as to the recovery and quality of the extracted signals of higher dimensions that arise in wide range of applications including medical image processing, seismic exploration, fluid mechanics, solutions of partial different equations, and compressed sensing.

REFERENCES

Badri, L. (2010), Development of neural networks for noise reduction, *The International Arab Journal of Information Technology*, 7(3), 156-165.

Bhosale, B., and A. Biswas (2013), Multi-resolution analysis of wavelet like soliton solution of KdV equation, *International Journal of Applied Physics and Mathematics*, 3(4), 270-274.

Bhosale, B., Ahmed, B., and A. Biswas (2013), Wavelet based analysis in bio- informatics, *Life Sciences Journal,* 10(2), 853-859.

Bhosale, B., Ahmed, B., and A. Biswas (2013), On wavelet based modeling of neural networks using graph theoretic approach, *Life Sciences Journal,* 10(2), 1509-1515.

Candes, E., and D. Donoho (2005), Continuous curvelet transform: Resolution of the wavefront set, *Applied And Computational Analysis,* 19 (2), 162–197.

Candes, E., Demanet, L., Donoho, D., and L. Ying (2006), Fast discrete curvelet transforms, multiscale modeling and simulation, *Applied And Computational Analysis,* 5(3), 861–899.

Cardoso, J. F., and A. Souloumiac (1993), Blind beam forming for non-gaussian signals, *IEE Proceeding* Part F, 140(6), 362-370.

Cornelis, J Stam, and J. Reijneveld (2007), Graph theoretical analysis of complex networks in brain, *Nonlinear Biomedical Physics,* **1**:3 doi:10.1186/1753-4631-1-3

Hassan, E., Laasri, A., Akhouayri, E., Agliz, D., and A. Atmani (2013), Article: Seismic signal classification using multi-layer perceptron neural network, *International Journal of Computer Applications,* 79(15), 35-43.

Hyvarinen, A., and E. Oja (2000), Independent Component Analysis: Algorithms and Applications, *Neural Networks,* 13(4-5), 411-430.

Jim, XU, and B. Zheng (2002), Neural networks and Graph theory, *Science in China (Series F)*, 45(1), 1-24

Kota, N., and G. Reddy (2011), Fusion based gaussian noise removal in the images using curvelets and wavelets with gaussian filter, *International Journal of Image Processing*, 5(4), 230-238.

Motwani, M., Gadiya, M., and R. Motwani (2004), Survey of image denoising techniques, Proceedings of global signal processing expo and conference, 27-30.

Moya, A., and K. Irikura, (2010), Inversion of a velocity model using artificial neural networks, *Computers & Geosciences,* 36, 1474–1483

Podolak, IT. (1998), Functional graph model of a neural network, *IEEE Transaction Systems*, 28(6), 876-81

Scarselli, F., Gori, M., Tsoi, A., Hagenbuchner, M., and G. Monfardini (2009), The graph neural network model, Neural Networks, *IEEE Transaction Systems*, 20(1), 61-80

Shehata, S., Diab, M., Salam, M., and M. Sayed (2013), Analysis of blind signal separation of mixed signals based on fast discrete curvelet transform, *International Electrical Engineering Journal*, 4 (4) 1140-1146.

Shimshoni, Y., and N. Intrator (1998), Classification of seismic signals by integrating ensembles of neural networks, *IEEE Transactions on Signal Processing*, 46(5), 45-56

Sivakumar, R. (2007), Denoising of computer tomography images using curvelet transform, *ARPN Journal of Engineering and Applied Sciences*, 2(1), 26-34

Valpola, H. (2000), Bayesian ensembel learning for non-linear factor analysis, *Acta   Ploytechnica, Scandinavica, Mathematics and Computing Series*, 108, 54-64.

Yaser, N., and Mahdi, J. (2011), A Novel Curvelet Thresholding Function for Additive Gaussian Noise Removal, International Journal of Computer Theory and Engineering, 3(4), 169-178.